Image Processing System based on similarity/dissimilarity measures to classify binary images from contour-based features

A Thesis Presented for the Master of Science in Electrical Engineering Degree

Student: Juan Sebastián Blandón Luengas

Director: Álvaro Ańgel Orozco Gutiérrez

Co-director: Andrés Marino Álvarez Meza.

Universidad Tecnológica de Pereira
Engineering Faculty - Electrical Engineering Program
Master in Electrical Engineering
Automatics Research Group
Pereira, Risaralda, Colombia
2020

Content

List of Figures

List of Tables

1 Symbols and Abbreviations

1.1 Symbols

In this document, we will refer to matrices using upper case bold letters and vectors as lower case bold letters. Parameters, vector positions, and iteration indices will be denoted as lower case letters. Other quantities definitions will be introduced in the next tables.

Table 1: Definitions from Sections 4.2.1, 4.2.2 and 5.2.2

Definitions from Sections 4.2.1 and 4.2.2			
Symbol	**Description**	**Symbol**	**Description**
$S(t)$	Shape	t	Position at contour shape
x_s, y_s	Shape coordinates	κ_c	curvature coefficient value
ξ_l	Step length		
Symbol	**Description**	**Symbol**	**Description**
$\mathcal{P}$	Probability distribution space	f	Function in RKHS
X, Y	Random variables	ζ_E	Similarity function in weighted `MMD`
$\mathcal{X}$	Mapping function	$\beta_{x'}$	Filter weight for x'
$\mathbb{P}, \mathbb{Q}$	Distribution functions	ω_n	Filter codebook
$\mu\{\cdot\}$	Marginal embedding operator	d_E	Euclidean distance
$\mathcal{H}$	RKHS	κ_G	Gaussian kernel
$\phi\{\cdot\}$	Embedding from $\mathcal{X}$ to $\mathcal{H}$	σ	Bandwidth value
$\kappa\{\cdot\}$	Characteristic kernel	e_t	Error tolerance
$\epsilon_{\mathbb{U}}$	Quantization size		
Definitions from Section 5.2.2			
Symbol	**Description**	**Symbol**	**Description**
$\boldsymbol{x}, \boldsymbol{x}'$	`CCS` or `CS`	r	A warping path size
N, M	`CCS` and `CS` lengths	L_k	Maximum warping path size
n, m	Position at `CCS` and `CS` lengths	K	Maximum number of warping paths
ζ	`DTW` algorithm localized function	γ	distance function between `CCS` and `CS`
$\boldsymbol{\Psi}$	`DTW` matrix	ψ_k	k-path cost
$\Psi_{n,m}$	`DTW` matrix elements	Q	number of features
$\boldsymbol{z}_k$	k-th warping path		

Table 2: Definitions from Sections 5.2.3, 6.2.1 and 6.2.2

Definitions from Section 5.2.3			
Symbol	**Description**	**Symbol**	**Description**
$\boldsymbol{I}_{n'}$	Image	$\boldsymbol{x}_{n'}, \boldsymbol{x}_{n''}$	`CSS` or `CS` for step length value ξ_l
W', H'	`CCS` and `CS` lengths	$\hat{\boldsymbol{F}}$	Feature space matrix for selected features
C	Total number of image labels	$\boldsymbol{f}$	Individual features from $\boldsymbol{F}$ and $\hat{\boldsymbol{F}}$
η	Total number of step lengths	Q'	Number of relevant features
$\boldsymbol{F}$	Feature space matrix	$\boldsymbol{\rho}$	ReliefF relevance vector
$\boldsymbol{D}^l$	Distance matrix	g	Nearest neighbors in ReliefF
$d^{(l)}_{n',n''}$	Elements from distance matrix at scale l	Ω	Set of ReliefF nearest neighbors

Definitions from Sections 6.2.1 and 6.2.2			
Symbol	**Description**	**Symbol**	**Description**
κ	`CS` value	G	Cluster centroid
T	Total number of vertices	$\mathbb{B}$	Mapping space of codebooks elements
$\boldsymbol{v}$	Vertices vector	$\boldsymbol{w}_{ij}$	Shape codes
τ	Threshold for vertices computation	$\boldsymbol{\pi}_{ij}$	Set of nearest neighbors in $\boldsymbol{B}$
$\mathcal{C}(S)$	Contour fragments set	l'	Regions from shape $S(t)$
c_{ij}	Contour fragment pair	$R_{r'}$	Subregions to apply max-pooling
$\boldsymbol{\theta}$	Context shape descriptor	ν	Number of subregions to apply max-pooling
d	c_{ij} dimension	$\Gamma_{r'}$	Max-pooling function
P	Reference points in $S(t)$	$\boldsymbol{X}_s$	Output matrix from BoCF descriptors
H	Histogram descriptor	N'	Number of images of the database
$\boldsymbol{B}$	Shape codebook	Q	Number of BoCF descriptors
$\boldsymbol{b}$	Shape codebook vectors	$\boldsymbol{x}_{s_{n'}}$	Vectors from BoCF output matrix

Table 3: Definitions from Sections 6.2.3 and 6.2.4

Definitions from Section 6.2.3			
Symbol	**Description**	**Symbol**	**Description**
$\boldsymbol{M}$	Data matrix from different approaches	β	Regularization parameter
$\boldsymbol{y}$	Image labels vector	$\hat{\beta}$	Regularization parameters estimation
ϵ	Upperbound of `LASSO` coefficients sum	λ	Penalization parameter
Definitions from Section 6.2.4			
Symbol	**Description**	**Symbol**	**Description**
$\boldsymbol{K}$	Kernel Matrix	$\boldsymbol{K}_c$	Centered kernel matrix
$\boldsymbol{K}_y y$	Kernel label matrix	$\|\cdot\|_F$	Frobenius norm
$\boldsymbol{K}_X X$	Kernel over input matrix	$\hat{\rho}$	Alignment measure
$\boldsymbol{a}$	Projection vectors	$\boldsymbol{W}$	Projection matrix
$\boldsymbol{I}$	Identity matrix		

1.2 Abbreviations

In the next table, we will show the abbreviations used in this document. We refer to methods using this fond type `method` and uppercase letters to depict simple abbreviations.

Table 4: Abbreviations from the whole text

Abb	**Description**	**Abb**	**Description**
IPS	Image Processing System	`ICS`	Integrating Skeleton and Contour paths
BI	Binary Image or Images	`NN`	Neural Network
SC	Shape Classification	`RNN`	Randomized `NN`
CSL	Colombian Sign Language	`SVM`	Support Vector Machine
FE	Feature Extraction	`BoW`	Bag of Words
FR	Feature Representation	`BoCF`	Bag of Contour Fragments
HD	High Dimensional	`BoSP`	Bag of Skeleton Paths
FS	Feature Selection	`BoSF`	Bag of Shape Features-based
DR	Dimensionality Reduction	`CS`	Curvature Sequence
`CCS`	Curvature Coefficient Sequence		

Table 5: Abbreviations from the whole text

Abb	Description	Abb	Description
DTW	Dynamic Time Warping	RKHS	Reproducing Kernel Hilbert Space
E	Exhaustive	WMMD	Weighted MMD
NE	Non-Exhaustive	KAF	Kernel Adaptive Filtering
LASSO	Least Absolute Shrinkage and Selection Operator	QKLMS	Quantized Kernel Least Mean Square
LDR	Linear DR	HMM	Hidden Markov Model
NLDR	Non-Linear DR	NC	Novelty Criterion
LD	Low Dimensional	EUC	Euclidean metric
PCA	Principal Component Analysis	SKL	Symmetric Kullback-Leibler metric
tSNE	t-Distributed Stochastic Neighbor Embedding	IDSC	Inner Distances Shape Context
SNE	Stochastic Neighbor Embedding	KeD	Kernel-edit Distance
JSE	Type 2 Mixture of Kullback-Leibler Divergences	LSS	Line Segment Statistics
NeRV	Neighbor Retrieval Visualizer	BoCF+BoSP	BoCF and Bag of Skeleton Paths
CKA	Centered Kernel Alignment	BoSCP	Bag of Skeleton-associated Contour Parts
DB	Databases	BoCF-LP	BoCF using a Learning Pooling function
SSLR	Static Sign Language Recognition	BoSCP-LP	BoSCP using a Learning Pooling function
KNN	K-Nearest Neighbor	CBoW	Curvature Bag of Words
MMD	Maximum Mean Discrepancy	EID	Enlacement and Interlacement Shape Descriptor
PR	Prototype Representation		

2 Introduction

2.1 Motivation

Image Processing Systems (IPS) try to solve tasks like image classification or segmentation based on its content. Many authors proposed a variety of techniques to tackle the image classification task. Plenty of methods address the performance of the IPS [1], as long as the influence of many external circumstances, such as illumination, rotation, and noise [2]. However, there is an increasing interest in classifying shapes from binary images (BI). Shape Classification (SC) from BI considers a segmented image as a sample (background segmentation [3]) and aims to identify objects based in its shape. There are broad ways to use SC and BI SC to solve specific tasks from different fields, such as the renewable energy industry, which used SC to determine roof shape. That is important since, according to roof morphology, they can design the photovoltaic cell to install [4]. In mechanical systems, they introduced SC to predict the wear states of a machine. They represent particles as a binary shape, whose morphology allows them to classify these particles preventing machine damages in industry [5]. Besides, medicine uses IPS to enhance its diagnosis procedures. In fact, regarding mammographic evaluation, BI SC allows them to identify masses and classify them as suspicious [6]. Also, BI SC supports hand recognition [7], object tracking [8], image sensing [9], among other applications.

Automatic Research Group tried to tackle the BI SC problem. This group developed a project named *Methodology to recognize and translate Colombian Sign Language (CSL) using vision computer techniques*. In this work, they tried to solve the sign discrimination process using machine learning techniques to identify BI shapes. As a result, they proposed a robust scheme that recognizes static signs from CSL. However, this task can be enhanced, since BI SC is a broad field which presents newer challenges.

2.2 Problem

IPS, based on BI, implements a set of stages to recognize shapes. First, a sensor takes pictures to recognize objects [10]. Afterwards, a Feature Extraction (FE) stage comes to get descriptors based on shape morphology features [11,12], curvature [13,14], among others. FE phase forces to decide: should we classify the features extracted to recognize BI shapes, or should we represent them, such as favors discrimination stage? This question does not have an answer since there is not a consensus about which strategy is better. Some authors classify the obtained descriptors [11, 15–19]. However, other authors implemented representation strategies to reveal patterns from their extracted features [20–24]. Thus, there is no evidence concerning a Feature Representation (FR) stage that enhances IPS performance.

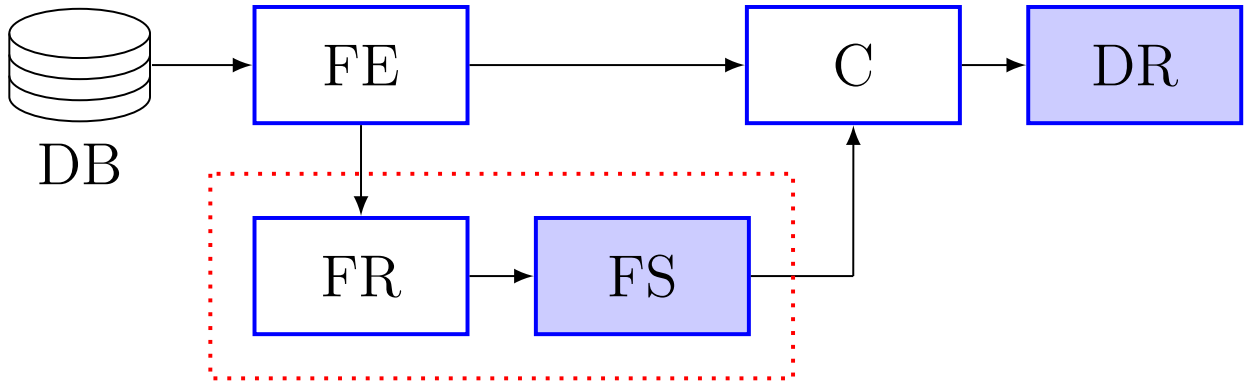

Figure 1: Flowchart of basic IPS pipeline. Red dotted rectangle represents a possible ensemble of methods to enhance IPS performance. Filled rectangles introduces new methods to IPS pipeline to gather and visualize relevant information.

Notwithstanding, the FE and FR stages can introduce high dimensional (HD) descriptors increasing the task complexity. This can lead to the feature space to keep redundant information since some strategies introduce scale information [20–25]. Besides, classify HD data can lead to the curse of dimensionality [26]. Hence, to avoid those problems, these methodologies should use Feature Selection (FS) techniques to prune redundant attributes obtained from FE and FR [27]. The IPS mentioned above also lacks a way to represent the relevant information. Then, Dimensionality Reduction (DR) takes an important role, since these techniques are useful to compress and visualize information [28]. Yet, most of the previous techniques do not implement any of the previous stages. The previous reasons show that there is an underlying problem that should be solved. In Fig. 1, the flowchart shows the ensemble of methodologies to tackle the problems mentioned.

2.3 Related Work

The State-Of-The-Art concerning BI SC might be described by two approaches, as we mentioned in Section 2.2: FE and FR. Regarding the first one, some methodologies try to classify shapes using specific methods to get descriptors. Authors in [11] stated a methodology were they computed inner distances along image shape boundaries, combining this characterization amongst multidimensional scaling, and short-path inner distances to recognize shapes. They showed that inner distance works for shapes with great articulation. However, it is sensitive to shapes under occlusion and large deformation. Authors in [12] proposed a method based on skeleton and contour features, which later fuses into a technique named Integrating Skeleton and Contour paths (`ICS`). Notwithstanding, their method needs closed shapes to get the descriptors, which is not always possible because of the BI quality. Authors in [16] proposed a contour-based representation using height functions as descriptors.

They alleged that these features are invariant to geometric transformations and insensitive to noise perturbations. Still, this feature is sensitive to local boundary deformations, which induces lose geometric information. Authors in [18] introduced a shape descriptor based on straight-line segment statistics, which is robust and tolerant against shape scale and orientation. However, this method fails when samples suffer occlusions. Authors in [19] explored bioinformatics tools to solve SC. They encode a 2D shape contour using the chain code. Then, they transform them into biological sequences using encoding strategies. From these sequences, they compute a similarity measure to get a representation matrix. Their method achieves excellent results over datasets which present high inter-class variability. But, their approach fails when this variability is low. Authors in [29] used a Neural Network (`NN`) as an FE method. To get the descriptors, they combine parameters for a Randomized `NN` (`RNN`), using the NN weights as features. To represent the BI, they choose the network configuration, which contains a low set of features without compromising system performance. However, since the `RNN` generates a non-linear combination of features, their method suffers from interpretability.

The second category of methodologies implements FR to enhance the FE stage. Authors in [15] proposed a symbol strings characterization, comparing each sample under a Kernel-edit distance, which allows them to classify using a Support Vector Machine (`SVM`). Yet, kernel methods are slow to train if the dataset has a great number of samples. Later, the author in [17] stated a Bag of Words (`BoW`) -like framework, which tries to integrate different shape-based and structure-based features. Notwithstanding, this technique does not capture scale information, which sometimes is useful to discriminate among shapes. In contrast, authors in [20] developed a descriptor called Bag of Contour Fragments (`BoCF`) based on the BoW model. This method decomposed each image into contour fragments, describing each of them using a shape descriptor, classifying through a linear `SVM`. The difference regarding work in [17] is that they applied pyramid pooling, which introduces scale information. Yet, the attribute domain is complex because the number of features is greater than the set of attributes got in [17], increasing system complexity. Hence, authors in [21] proposed a framework that joints `BoCF` and Bag of Skeleton Paths (`BoSP`), trying to get contour and structure-based information. Still, they increase the complexity using almost twice of features of the `BoCF` method. In [25] they state a technique which classify shapes considering scale, rotation and viewpoint variations. Yet, the method fails because some classes from the databases they used were ambiguous, leading the system to fail. Authors in [23] and [24] proposed `BoCF` and Bag of Shape Features-based (`BoSF`) methods to learn an adaptive pooling function. These methods reach significant performance results but present the same problem as [20]; the feature space complexity increases because they got a great number of features. This lead to got ambiguous results because some of the datasets used had low inter-class variability. In the work of [30], the authors proposed a Curvature Bag of Words model. This method differs from BoCF since it can deal with shape deformations. However, it keeps the same problem as the other BoCF-based methods, which are the high complexity

because the number of features gets once they applied the FE method. Recent works present alternatives to shape-based methods. Authors in [31], proposed a characterization method based on enlacement and interlacement of directional spatial relations. They make a pair-wise comparison about concavity of two objects, and also, about how a shape enlaces to its surrounding background. However, its problem relies on the loss of relevancy when the number of classes grows drastically, getting combined with other kinds of descriptors.

2.4 Proposal

As we exposed, most schemes propose an ensemble of FE and classification [11,12,16,18,19,29] or FE, FR, and classification [15, 17, 20, 23–25, 30, 31] (See Fig. 1). However, those FE and FR strategies present problems like interpretability and redundancy. Thus, we proposed an IPS based on similarity/dissimilarity measures to reveal relevant patterns in BI classification tasks from image coding strategies. This ensemble of methodologies ensured to be invariant to scale, noise, transformations, and occlusions of the input data. To do that, we extracted curvature-like features [13,20]. We used Curvature Coefficients Sequences (`CCS`) and Curvature Sequences (`CS`) since they are simpler than other curvature extraction methods like discrete contour evolution [30]. To ensure that these attributes capture useful information, we implement the similarity/dissimilarity-based algorithm named Dynamic Time Warping (`DTW`) [32]. This way to depict data allows us to test if a sample is more informative than others, no matter if the sample has local distortions. Besides that FR strategy, we implemented `BoCF`, since this method keeps scale information and is invariant to a different type of perturbations. The abovementioned methods cover FE and FR stages, but as we exposed in Section 2.2, there are other challenges to beat. Hence, we introduce here FS as a phase that will allow us to avoid the redundancy that features get from `BoCF` exposes [20, 23, 24, 30]. Then, to prune redundant information, we implemented two kinds of techniques: Exhaustive (E) and Non-Exhaustive(NE). These types of approaches allow us to test our system under parametric and non-parametric selection techniques. In this work, we used as E a supervised selection technique named ReliefF [33] because, in SC task, we have labels. As the NE method, we implemented the Least Absolute Shrinkage and Selection Operator (`LASSO`) [34], which allows us to discard features according to binary regression coefficients. However, we have not solved the lack of interpretability. Thus, we introduce DR to refine our ensemble since it is used to improve our classification stage and to visualize the spaces that our methodology found. To implement DR, we test various methods, since Linear DR (LDR) to Non-Linear (DR). Both approaches map data from HD space to Low-Dimensional (LD) one. In this work we explore LDR methods like Principal Component Analysis (`PCA`), and NLDR like t-Distributed Stochastic Neighbor Embedding (`tSNE`) [35], Stochastic Neighbor Embedding (`SNE`) [36], Type 2 Mixture of Kullback-Leibler Divergences (`JSE`) [37], Neighbor Retrieval Visualizer (`NeRV`) [38], and Centered Kernel Alignment (`CKA`) [39]. Those methods allow us to explore how our methodology solves

inter-class and intra-class variability of the Databases (DBs). To prove our IPS, we applied it over three State-Of-The-Art DBs that have high inter-class variability: `Kimia 99-Shape` Database, `Animals Dataset`, and `MPEG-7_CE-Shape-1 Part B`. Besides these datasets, we applied our methodology over a fourth database named `Polish Hand Gesture Recognition` DB, which describes a real-world problem: Static Sign Language Recognition (SSLR). To infer shapes, we used different classifiers: K-Nearest Neighbor (`KNN`), Linear, Radial base function `SVM`, Linear `SVM`, and Logistic Regression. We find that our methodology answers the question about when it is useful to use FE without FR and when to include it. Also, it solves BI SC challenges like the lack of interpretability of the results and the introduction of redundancy once we applied FR stages. Furthermore, our IPS can support SSLR under controlled conditions but fails when the homogeneity of the among signs is high.

The remainder of this text is organized as follows: Section 3 presents the general and specific objectives of this work, Section 4, Section 5, and Section 6 describes methods, experimental set-up, results and discussion from each specific objective, Section 7 presents the conclusions, Section 8 describes the future work, Section 9 expose the academic results from this investigation, and Section 10 shows the acknowledgments.

3 Objectives

3.1 General Objective

To develop an image processing scheme based on similarity/dissimilarity measures to reveal relevant patterns in binary image classification tasks.

3.2 Specific objectives

1. To develop a characterization methodology based on shape coding to support object classification tasks over binary images.
2. To develop a representation strategy based on similarity/dissimilarity measures, which quantifies similarities between regions of interest to code relevant patterns in binary images.
3. To develop a dimensionality reduction methodology based on similarity/dissimilarity measures that allow coding relevant patterns in binary image classification and data visualization tasks in image processing systems.

4 KAF-based BI classification from curvature descriptors

4.1 Introduction

The first specific objective aims to develop an FE strategy based on shape coding to support BI SC. This problem is challenging because the unique information in the BI is the shape object, and the background represented as a black area. Thus, as we exposed in Section 2.3, some authors proposed approaches like a skeleton-based and contour-based FE methods. However, one of the major problems of BI is their tendency to distortion, from occlusions to segmentation. Hence, both FE approaches are vulnerable to these types of perturbation. Yet, contour-based techniques are simpler and more powerful, than a skeleton-based [13]. Thus, we want that simplicity to prevail in our IPS.

To develop the FE technique we implemented `CCS` extraction as [13] proposes. Notwithstanding, we implement kernel-based filters to represent sequential data from BI. This method learns a model from each `CCS` sequence using adaptive kernels. Therefore, we can apply FR methods to compare each model in that mapped space since, in that space, each class can become separable. Thus, the keys in our proposal rely on the FE implementation of Kernel Adaptive Filters (`KAF`) and developing a Maximum Mean Discrepancy-based (`MMD`) metric to compare each model sample. This section will show the details of the proposed methodology to solve our first specific objective.

4.2 Methods

4.2.1 Feature Extraction: Parametric Curvature-based Descriptors

To represent the contour shapes, we proposed a FE stage that obtains sequential descriptors from BI. Once we extract the contours using edge filters (Canny filter), we compute two types of sequential features: `CCS` and CS. Both are defined as follows:

Curvature Coefficient Sequences Let $S(t)$ be a trajectory vector from an image edge. Each element from vector $S(t) = s(t)_1, s(t)_2, ..., s(t)_{L'}$, where $S(t) \in \mathbb{R}^{L'}$,$t \in [0, 1]$ is a trajectory point, and L' are the total number of trajectory points of $S(t)$. An element $s(t)_{L'}$ has also a cartesian coordinates representation given by $s(t)_{L'} = (x_s, y_s)$ [40].

One can define a curvature coefficient value from two adjacent vectors of length ξ_l as:

$$\kappa_c = \arccos\left(\frac{(s(t)_{L'} - s(t)_{L'-\xi_l}) \cdot (s(t)_{L'+\xi_l} - s(t)_{L'})^\top}{\|s(t)_{L'} - s(t)_{L'-\xi_l}\|_2 \|s(t)_{L'+\xi_l} - s(t)_{L'}\|_2}\right) \tag{1}$$

where $\kappa_c \in \mathbb{R}^+$ and $\xi_l > 0 \in \mathbb{Z}$ is the step length parameter. κ_c is a number which represents how much concave or convex a curve being. This type of representation is parametric, since depends on the ξ_l parameter. However, it does not offers information about curvature direction.

4.2.2 Feature Representation: KAF-based representation using MMD Enhancement

To compare two different objects from their curvature sets $\{x_n\}_{n=1}^N$ and $\{y_m\}_{m=1}^M$, we introduce a Reproducing Kernel Hilbert Space Embedding-based distance as follows: let $\mathcal{P}$ be the space of all probability distributions and let $X, Y \subset \mathcal{X}$ be two random variables that follow the distribution functions $\mathbb{P}$ and $\mathbb{Q}$, respectively; then $\mathbb{P}, \mathbb{Q} \in \mathcal{P}$, $x_n \in X$, and $y_m \in Y$. Let $\mu\{\cdot\}$ be a marginal embedding operator mapping a given sample $x \in X$ from a probability distribution $\mathbb{P}$ to a Reproducing Kernel Hilbert Space (RKHS) $\mathcal{H}$, as follows: $\mu(\mathbb{P}) = \mathbb{E}_X[\phi(x)] = \int_{\mathcal{X}} \phi(x) d\mathbb{P}(x)$, where $\phi : \mathcal{X} \to \mathcal{H}$. This embedding of probability distributions into RKHS allows us to compute distances between them. According to [41], the RKHS-based distance over the probability measures $\mathbb{P}$ and $\mathbb{Q}$, yields: $d_\kappa^2(\mathbb{P}, \mathbb{Q}) = \|\mu(\mathbb{P}) - \mu(\mathbb{Q})\|_{\mathcal{H}}^2$, which can be rewritten as: $d_\kappa^2(\mathbb{P}, \mathbb{Q}) = \left\|\int_{\mathcal{X}} \phi(x) d\mathbb{P}(x) - \int_{\mathcal{X}} \phi(y) d\mathbb{Q}(y)\right\|_{\mathcal{H}}^2$.

Afterward, we define a function $\kappa(x, x') = \langle \phi(x), \phi(x') \rangle_{\mathcal{H}}$, $\forall x, x' \in \mathcal{X}$ as a reproducing characteristic kernel on $\mathcal{H}$. If the probability distributions $\mathbb{P}(x)$ and $\mathbb{Q}(y)$ admit density functions $p(x)$ and $q(y)$, respectively, we have $d\mathbb{P}(x) = p(x)dx$ and $d\mathbb{Q}(y) = q(y)dy$. Also, the density functions $p(x)$ and $q(y)$ can be computed as a dot product between a function $f \in \mathcal{F}$ and the mapping φ into the RKHS $\mathcal{F}$, e.g, $p(x) = \langle f, \varphi(x) \rangle_{\mathcal{F}} = \int_{\mathcal{X}} \alpha_{x'} \varphi(x) dx'$, where $\varphi : \mathcal{X} \to \mathcal{F}$ and $\alpha_{x'} \in [0, 1]$. Next, we rewrite $d_\kappa^2(\mathbb{P}, \mathbb{Q})$ in terms of the α parameters as follows [42]:

$$\begin{aligned} d_\kappa^2(\mathbb{P}, \mathbb{Q}|\alpha_x, \alpha_y) &= \int_{\mathcal{X}} \int_{\mathcal{X}} \alpha_x \alpha_{x'} \kappa(x, x') \zeta(x, x') dx dx' + \int_{\mathcal{X}} \int_{\mathcal{X}} \alpha_y \alpha_{y'} \kappa(y, y') \zeta(y, y') dy dy' \\ &- 2 \int_{\mathcal{X}} \int_{\mathcal{X}} \alpha_x \alpha_y \kappa(x, y) \zeta(x, y) dx dy, \end{aligned} \tag{2}$$

where $\zeta(x, x') = \langle \varphi(x), \varphi(x') \rangle_{\mathcal{F}}$. The expression in (2) can be seen as a weighted enhancement of the well-known `WMMD` with regard to the mapping function φ in the $p(x)$ and $p(y)$ estimation. In this sense, to code relevant curvature patterns, we propose to learn the weights in `WMMD` towards a Kernel Adaptive Filtering (`KAF`) technique. Namely, the quantized kernel least mean square (`QKLMS`) algorithm is selected as a straightforward solution [43]. So, the densities

are computed from an input-output pair of samples $\{x_{n+1}, x_n\}_{n=1}^{N-1}$ and QKLMS predictions: $\hat{x}_{n+1} = \sum_{x' \in \Omega_n} \beta_{x'} \zeta(x_{n+1}, x')$, being Ω_n the filter codebook at the n-th iteration and $\beta_{x'} \in \mathbb{R}$. Therefore, QKLMS estimates the curvature sequence into a RKHS following a Markovian constraint to preserve the spatial dependencies of the object. Besides, the QKLMS includes a novelty criterion (NC) based on the euclidean distance $d_E(x_n, \Omega_{n-1})$, with $i \in \{1, 2, \ldots, N\}$. Later, given two curvature sequences from X and Y, that is, $\{x_n\}_{n=1}^{N} \sim \mathbb{P}$ and $\{y_n\}_{m=1}^{M} \sim \mathbb{Q}$, the densities are computed as: $\hat{p}(x) = \sum_{x' \in \Omega_{N-1}^x} \hat{\alpha}_{x'} \zeta(x, x')$ and $\hat{p}(y) = \sum_{y' \in \Omega_{N-1}^y} \hat{\alpha}_{y'} \zeta(y, y')$. Furthermore, to preserve the metric properties in WMMD, the weights are normalized as: $\hat{\alpha}_{x'} = |\beta_{x'}| / \sum_{\beta_{x'} \in \Omega_{N-1}^x} |\beta_{x'}|$ and $\hat{\alpha}_{y'} = |\beta_{y'}| / \sum_{\beta_{y'} \in \Omega_{M-1}^y} |\beta_{y'}|$. Finally, using a Gaussian kernel $\kappa_G(\cdot, \cdot) \in \mathbb{R}^+$ to infer the RKHSs, the WMMD is written as:

$$\begin{aligned} \hat{d}_\kappa^2(\mathbb{P}, \mathbb{Q} | \alpha_x, \alpha_y) &= \sum_{x, x' \in \Omega_{N-1}^x} \alpha_x \alpha_{x'} \kappa_G(x, x' | \sigma + 2\sigma_x) + \sum_{y, y' \in \Omega_{M-1}^y} \alpha_y \alpha_{y'} \kappa_G(y, y' | \sigma + 2\sigma_y) \\ &\quad - 2 \sum_{x \in \Omega_{N-1}^x, y \in \Omega_{M-1}^y} \alpha_x \alpha_y \kappa_G(x, y | \sigma + \sigma_x + \sigma_y), \end{aligned} \tag{3}$$

where $\sigma, \sigma_x, \sigma_y \in \mathbb{R}^+$ are the bandwidth values for the characteristic and the QKLMS kernels, respectively.

4.3 Experimental Set-up

4.3.1 Databases

To assess the WMMD we used two databases (DB): the Kimia 99-Shape Database [1] (Kimia 99 Shape DB), and the MPEG-7_CE-Shape-1 Part B [2] - (MPEG7 DB), holding 9 and 13 classes, respectively. Both datasets comprise binary images with different resolutions and perturbations, e.g., occlusion and rotation. We sampled MPEG7 DB into 13 classes as [44] suggested. Fig. 2 shows the DBs used in this proposal.

4.3.2 Quality Assessment

Given a binary image $\boldsymbol{I}_{n'} \in \mathbb{R}^{W' \times H'}$ coding a shape, we carried out a median filtering procedure to clean the object shape. Thus, we applied a Canny filter to highlight the object outline. Afterward, we run some morphological operations, i.e., opening, closing, and erosion, to prevent non-closed trajectories. Then, we fitted a Savitzky-Golay [45] filter over those trajectories to exclude labeling errors. Then, to quantify shape variations, we compute a

[1] https://vision.lems.brown.edu/content/available-software-and-databases
[2] http://www.dabi.temple.edu/~shape/MPEG7/dataset.html

(a) `Kimia 99 Shape` DB

(b) `MPEG7` DB

Figure 2: Samples from each DB used in method.

curvature coefficients vector $\boldsymbol{x} \in \mathbb{R}^N$ holding N coefficients based on first-order changes [13]. To compute those `CCS`, we extracted those trajectories from the farthest horizontal point to the right of the shape centroid, getting each curvature point from this first coordinate in a counter-clockwise manner ruled by a step length $\xi_l \in \mathbb{N}$. We used an equally spaced grid from 3 to 20 since a greater value of ξ_l does not provide any extra information. To tackle the BI SC, we proposed to compute filter predictions based on each `CCS` got in Section 4.2.1. To do that, we train multiple `QKLMS` filters. Concerning the `QKLMS` parameters, a grid of 100 values is built, varying $\epsilon_{\mathbb{U}}$ from 1 to 10, while we computed e_t based on the input data standard deviation. We fixed the kernel bandwidth σ_x according to a Parzen-based density estimation of each curvature sequence. Further, we tuned the characteristic kernel bandwidth σ in terms of the classification accuracy, building a grid of 5 points from 0.003 to 0.3 in `Kimia 99 Shape` DB, and from 0.001 to 0.03 in `MPEG7` DB. We set the `QKLMS` learning rate as 0.9 for both experiments [46]. Once we trained the set of filters for every DB, we computed `WMMD`-based distances among shapes. That allows us to code in a similarity/dissimilarity-based manner the computed estimations for subsequent classification. We implement a 1-NN classifier from the `WMMD`-based distances obtained from Section 4.2.2. For concrete testing, we carried out a training-testing validation fixing the training set as the 80% and 50% of the input images from `Kimia 99 Shape` DB and `MPEG7` DB, respectively.

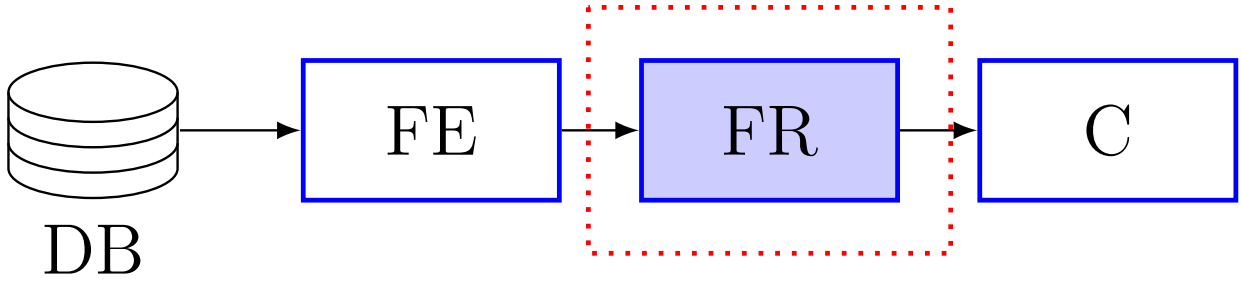

Figure 3: This flowchart presents the KAF method to represent curvature descriptors for BI classification. The blue filled rectangle represents the stage, which was modified with our WMMD enhancement..

4.3.3 Method Comparison

As a baseline, we test two State-Of-The-Art techniques: an HMM classifier based on the highest likelihood as discussed in [44], and a 1-NN classifier from an MMD representation [9]. Fig. 3 shows the flowchart of the ensemble of the previous techniques exposed techniques.

4.4 Results and Discussion

Left Fig. 4 shows a data sample contour plot from Kimia 99-Shape DB. As seen, points holding high curvature coefficient values (see color intensities) are related to extremities, which has greater morphological changes concerning the rest of the shape. Besides, from marker sizes, the QKLMS-based relevance analysis can code the main shape variations. Now, right Fig. 4 shows the filter prediction against the target curvature sequence. Curvature beginning is labeled by x_0, and as was exposed in Section 4.3.2, following counter-clockwise, it is found the midpoint of the sequence. In general, the KAF-based adjustment is low concerning the curvature predictions. Since our goal is to classify shapes towards a relevant curvature representation, the codebook and the filter weights are more appropriate for further discrimination task, e.g., the 1-NN-based C, than for the curvature value prediction. So, the training stage requires a trade-off between classification and filtering fitting.

In addition, Table 6 shows the results of classification for the Kimia 99-Shape DB. Overall, the accuracy obtained in this experiment was 77.8%, which is lower than the HMM and MMD approaches. However, in Quadrupeds, Humans, Hands, Rays, Rabbits, and Wrenches classes, our proposal achieves the highest performances. Moreover, our methodology fails for the Airplanes class; this ought to sharp changes along its contour and low curvature sequences in comparison to the other classes, which induces biased prediction results in the QKLMS filter. Further, Table 7 presents the classification results for the MPEG7 DB. This experiment was

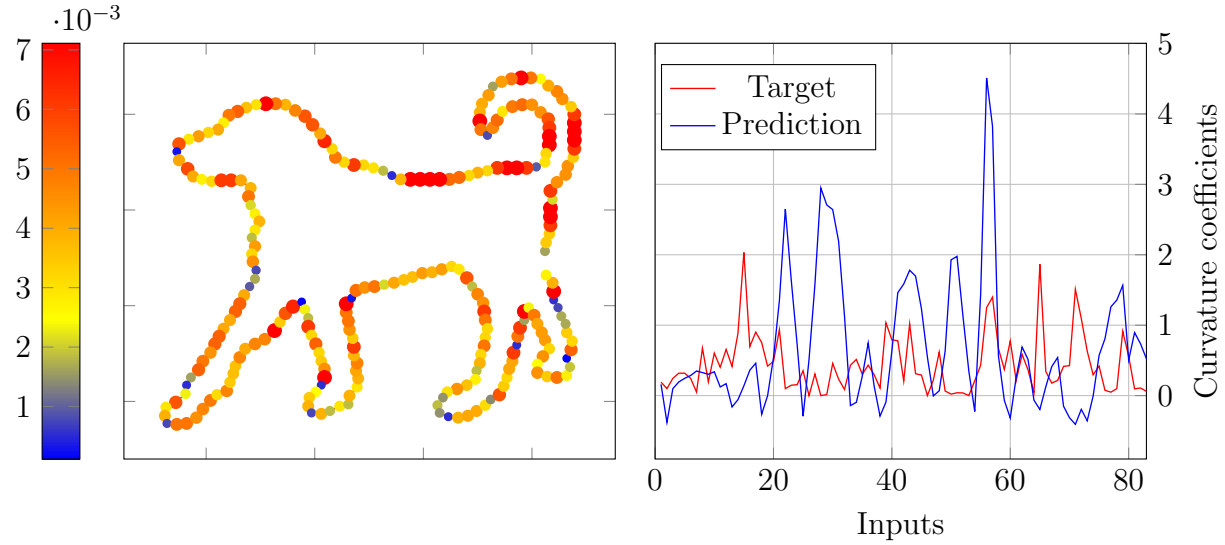

Figure 4: Left: $x_{n}{}_{n=1}^{N}$ for image 73 from `Kimia 99-Shape` DB. Right: QKLMS curvature prediction

successful since our methodology overcomes `MMD` and `HMM` schemes. However, for Shoe and Bone classes, their outcomes could be due to their class heterogeneity. For Bottle, Children, Flatfish, and Fountain classes, `WMMD` allows discriminating each of them properly.

In general, `WMMD` is competitive with other State-Of-The-Art methods. Previous results allow us to say that our `WMMD`-based methodology performs better than `HMM` models proposed by [44]. This outcome could be possible ought to our method introduces the NC (see Section 4.2.2) [43], which avoids using redundant curvature sequences. Moreover, these input filters are embedded in an RKHS, which allows us to maps non-linear shape structures.

4.5 Summary

We presented a novel approach to classify shapes based on sequential features extracted from contours and a `KAF`-based `MMD` enhancement. Our methodology extract contour-based attributes to apply later kernelized filters over these sequences. Kernel filters allow mapping the input data trough non-linear strategies. The previous FR technique favors class discrimination since our similarity/dissimilarity metric captures the relevant information associate with each BI. Besides, our approach captures non-linear curvature dependencies to code relevant shape patterns from BI, exposing sharp changes related to greater curvatures of each trajectory. We proposed to support object classification tasks over images, and our methodology achieves it since we used this contour-based information to recognize shapes from `Kimia 99 Shape` DB and the sampled `MPEG7` DB. Our results are comparable

Table 6: `Kima 99-Shape` DB classification results specific objective 1.

Class	**Proposal** TP(%)	**HMM** TP(%)	**MMD** TP(%)
Quadrupeds	100.0	100.0	50.0
Humans	100.0	100.0	100.0
Airplanes	0.0	81.8	50.0
Grebes	50.0	100.0	50.0
Fish	50.0	72.7	100.0
Hands	100.0	90.9	100.0
Rays	100.0	90.9	100.0
Rabbits	100.0	81.8	100.0
Wrenches	100.0	72.7	100.0
Accuracy(%)	**77.8**	87.9	83.3

Table 7: `MPEG-7` DB classification results specific objective 1.

Class	**Proposal** TP(%)	**HMM** TP(%)	**MMD** TP(%)
Bone	60.0	100.0	60.0
Glas	100.0	100.0	100.0
HCircle	90.0	80.0	80.0
Heart	90.0	90.0	70.0
Misk	100.0	80.0	100.0
Apple	70.0	80.0	60.0
Bottle	100.0	70.0	90.0
Children	100.0	90.0	90.0
Face	100.0	90.0	90.0
Flatfish	100.0	80.0	90.0
Fountain	100.0	80.0	90.0
Shoe	50.0	60.0	90.0
Teddy	80.0	90.0	100.0
Accuracy(%)	**87.7**	83.8	85.4

concerning the other State-Of-The-Art techniques because the performance achieve captures the inter-class variability from each DB. Thus, our method finds a different manner to get silhouette-based features since we represent each sample as a contour sequence representing the significant silhouette changes.

5 Similarity/dissimilarity-based representation for BI discrimination

5.1 Introduction

The second specific objective proposes to develop a similarity/dissimilarity-based representation that captures similarities between regions of interest from curvature-based features. It is alleged that it allows us to code discriminant patterns in BI and favor BI classification. This proposed objective aims to complement the first specific objective. Until here, the methodology explores how contour-based features and its subsequent representation favors the classification performance of BI. But, as we exposed in Section 2, in an IPS is usual to get a great number of features, being more notorious in the BI case. This might introduce redundancy when the techniques uses scale information [20–25].

To develop an FR based on similarity/dissimilarity measures, we used the known algorithm `DTW` [32], which allows us to compare local and globally two pairs of sequences. That is possible since we use the same FE strategy as Section 4.2.1 that computes a set of features related to different curvature scales. Hence, we propose to represent those features in an HD space and then applied FS to discard repetitive information. In this chapter, we will expose our modified methodology to support `SSLR`.

5.2 Methods

5.2.1 Feature Extraction: Curvature-based Descriptors

To represent the contour shapes, we used the same scheme as in Section 4.2.1. This FE stage gets `CCS` sequential descriptors from BI varying the ξ_l parameter, which leads to a set of scale-based features.

5.2.2 Feature Representation: DTW-based representation of contour information

To compare two different `CCS` $\boldsymbol{x} \in \mathbb{R}^N$, $\boldsymbol{x'} \in \mathbb{R}^M$ extracted from a pair of binary images, we use a Dynamic Time Warping-based distance (`DTW`). In fact, the `DTW` allows comparing two sequences of different length based on a localized function: $\zeta : \mathbb{R}\times\mathbb{R} \rightarrow \mathbb{R}^+$. So, if $\zeta(x_n, x'_m)$ is small $(n \in \{1, 2, \ldots, N\}, m \in \{1, 2, \ldots, M\})$, then x_n and x^p_m code a similar

curvature pattern; instead, if $\zeta(x_n, y_n)$ is high, they are dissimilar. Here, the Euclidean and Kullback Leibler-based localized functions are deemed as follows:

$$\zeta_{EUC}(x_n, x'_m) = \sqrt{(x_n - x'_m)^2}, \tag{4}$$

$$\zeta_{SKL}(x_n, x'_m) = (x_n - x'_m)(\log(x_n) - \log(x'_m)). \tag{5}$$

Further, a matrix $\boldsymbol{\Psi} \in \mathbb{R}^{N \times M}$, holding elements $\Psi_{nm} = \zeta(\boldsymbol{x}, \boldsymbol{x'})$, can be built by fixing the localized function ζ. The DTW algorithm searches the alignment between CCS as the minimum mean cost in $\boldsymbol{\Psi}$. Namely, a warping path set $\{\boldsymbol{z}_k \in \mathbb{N}^{2 \times L_k}\}_{k=1}^{K}$ is built, where $\boldsymbol{z}_k = [(n_1, m_1), \ldots, (n_{L_k}, m_{L_k})]$ is the k-th path of size $L_k \in \mathbb{N}$ into the matrix $\boldsymbol{\Psi}$, $n_r \in \{1, 2, \ldots, N\}$ and $m_r \in \{1, 2, \ldots, M\}$ depict the sample positions in vectors $\boldsymbol{x}$ and $\boldsymbol{x'}$, respectively, and $r \in \{1, 2, \ldots, L_k\}$. Notably, each path $\boldsymbol{z}_k$ satisfies the boundary, monotonicity, and step size conditions. Besides, a distance function $\gamma : \mathbb{R}^N \times \mathbb{R}^M \rightarrow \mathbb{R}^+$ between CCS can be defined as follows: $\gamma(\boldsymbol{x}, \boldsymbol{x'}) = \min_{k \in K} \psi_k(\boldsymbol{x}, \boldsymbol{x'})$, where $\psi_k(\boldsymbol{x}, \boldsymbol{x'}) = \sum_{r=1}^{L_k} \zeta(x_{n_r}, x'_{m_r})$ is the cost for the path k [32].

5.2.3 Feature Selection: ReliefF applied to contour-based features

Let $\{\boldsymbol{I}'_n \in \mathbb{R}^{W' \times H'}, y_{n'} \in \{1, 2, \ldots, C\}\}_{n'=1}^{N'}$ be a set holding N' binary images and output labels regarding C static signs. Moreover, let $\{\xi_l \in \mathbb{N}\}_{l=1}^{\eta}$ be a set of step length values used to reveal different curvature relationships within a binary image, a feature space matrix $\boldsymbol{F} \in \mathbb{R}^{N' \times Q}$ is built as: $\boldsymbol{F} = [\boldsymbol{D}^{(1)}, \boldsymbol{D}^{(2)}, \ldots, \boldsymbol{D}^{(\eta)}]$, where $\boldsymbol{D}^{(l)} \in \mathbb{R}^{N' \times N'}$ is a DTW-based matrix with elements: $d^{(l)}_{n'n''} = \gamma(\boldsymbol{x}_{n'}(\xi_l), \boldsymbol{x}_{n''}(\xi_l))$, $\boldsymbol{x}''_n(\xi_l) \in \mathbb{R}^{N'_{n'l}}$ is the CCS for the step length value ξ_l, and in this case $Q = N' \times \eta$. In practice, $\boldsymbol{F}$ holds a huge number of attributes (features). This fact can yield to overfitting in further classification stages, being necessary to infer a matrix $\hat{\boldsymbol{F}} \in \mathbb{R}^{N' \times Q'}$ ($Q' \leq Q$), with the most Q' relevant features favoring the trade-off between system complexity and SSLR performance. To do this, we compute the contribution of each feature in $\boldsymbol{F}$ in terms of the supervised information given by the labels in a relevance vector $\boldsymbol{\rho} \in \mathbb{R}^Q$ based on the Relief-F approach as [33]:

$$\rho_q = \frac{1}{N'} \sum_{n'=1}^{N'} \left(-\frac{1}{g} \sum_{\boldsymbol{f}_{n''} \in \Omega_{n'}^{y_{n'}}} \|\boldsymbol{f}_{n'} - \boldsymbol{f}_{n''}\|_2 + \frac{1}{g} \sum_{g \neq y_{n'}} \frac{p(y=c)}{1 - p(y=y_{n'})} \sum \|\boldsymbol{f}_{n'} - \boldsymbol{f}_{n''}\|_2 \right), \tag{6}$$

where $\Omega^c_{n'} = \{\boldsymbol{f}_{n''} : n'' = 1, 2, ..., g\}$ holds the g-nearest neighbors of $\boldsymbol{f}_{n'}$, and $p(y = c)$ is the probability that a sample belongs to the c-th class ($c \in \{1, 2, ..., C\}$), and $q \in \{1, 2, ..., Q\}$. In this manner, the higher the ρ_q value the better the q-th feature for discriminating hand gestures in SSLR. Therefore, a performance curve (in terms of classification accuracy) can be used to find the Q' most relevant features by adding one by one the input attributes concerning the ρ values.

Figure 5: PHGR DB used from the specific objective 2.

5.3 Experimental Set-up

5.3.1 Databases

To test the proposed DTW-based representation, we use the publicly available DB Polish Hand Gesture Recognition[3] - (PHGR DB), which is composed by 899 binary images representing 25 different hand gestures. Each sample, who have different resolutions, corresponds to a static PSL. For concrete testing, six classes and 25 samples per class are used as suggested by [47], since those samples were well discriminated in their work. Fig. 5 presents the PHGR DB from our selected samples.

5.3.2 Quality Assessment

We computed a feature space holding $N' = 150$ inputs and $Q = 2700$ attributes. Those 2700 features correspond to the 18 scales, and every scale describes a DTW-based distance matrix $\boldsymbol{D}^l$. Hence, we got a concatenated feature matrix, which will be the input for our FS phase.

To reduce the computed feature space matrix $\boldsymbol{F}$, we applied ReliefF to get the most representative attributes. We implemented the E FS method in our methodology because we wanted to determine whether or not a scale was informative. To do that, we fixed the number of neighbors for ReliefF at $g = 1$. Hence, we got a vector $\boldsymbol{\rho}$, which comprises information about whether a set of features was relevant. This stage requires an ensemble with a classifier to assess each descriptor subset performance.

To assess the methodology performance, we implement a nested cross-validation strategy to train a Linear classifier, a KNN, and a SVM. We used a Hold-Out scheme of $(70 - 30)\%$ training-testing, which we repeated ten times to get statistical relevance. We searched the

[3] http://sun.aei.polsl.pl/~mkawulok/gestures/

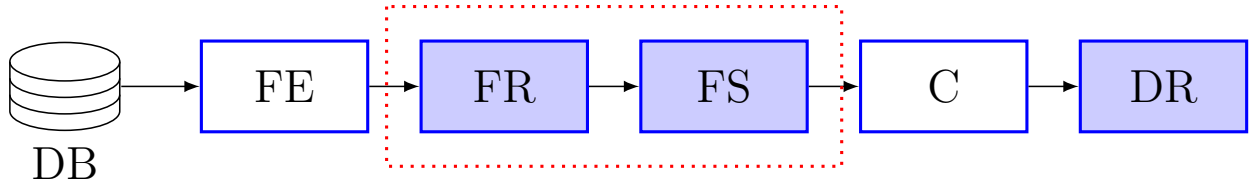

Figure 6: Flowchart of feature representation based on similarity/dissimilarity measures for BI discrimination. Blue filled rectangles represent the stages we add and modify to enhance the BI SC performance and interpretability.

number of neighbors for KNN in a grid from 1 to 15; we used a Gaussian Kernel for the SVM finding the bandwidth between $0.1\sigma_o$ to σ_o, being $\sigma_o \in \mathbb{R}^+$ the median of the Euclidean distances between input samples, and the regularization value between 0.1 to 1000.

To gather a visual representation of the reduced DB once we applied FS, we used Type 2 Mixture of Kullback-Leibler Divergences (JSE) as a visualization strategy. This method is an NLDR technique that is useful when the databases present non-linearity. Sampled PHGR DB notes non-linearity since some images share morphological dispositions that our strategy could not learn. To preserve local information about the categories, we fixed the parameter of weighting between LD and HD spaces at 0.95. Fig. 6 presents the set of methods used to develop the previous methodology.

5.4 Results and Discussion

Fig. 7 shows the curvature estimation of a preprocessed sample of class 4 in the PHGR DB, fixing the curvature step length at $\xi_l = 15$. Each point depicts a curvature coefficient computed, as explained in Section 5.2.1. The plot demonstrates that the CCS represents the main patterns concerning the shape variations, where bluish colors denote low curvature values and reddish the higher ones, which are related to smooth and sharp changes, respectively. Accordingly, the proposed features take essential information described by the morphological structure from the hand gesture.

Now, Fig. 8 shows the classification results obtained by adding one-by-one the sorting features using the ReliefF ranking. Remarkably, the EUC metric shows a better performance than the SKL. This outcome ought to SKL metric does not code the main differences between classes accurately, leading to poor performance results (See Fig. 8, the graph on the right). On the other hand, the EUC captures the relevant differences among CCS, generating a dissimilarity representation which accurately describes the SSLR patterns. Respecting to classifier performances, the Linear and the SVM show stable results for the two types of dissimilarity spaces. However, KNN does not increase its outcomes using a few attributes,

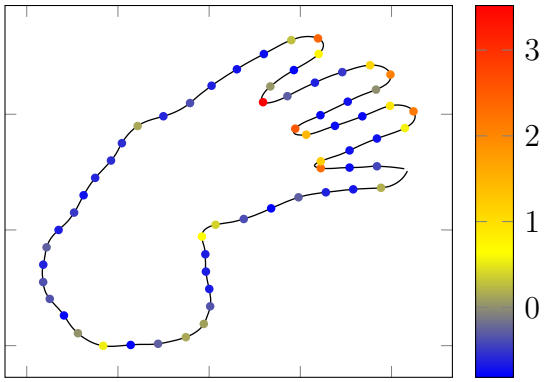

Figure 7: Curvature coefficient sequence for a sample in class 4 from PHGR DB.

which means that no matter how many features we used in the classifier can not improve its performance. According to the these results, the highest accuracies for the EUC and SKL are $84.79 \pm 2.95\%$ (851 features using the Linear classifier) and $72.29 \pm 5.11\%$ (291 features using the SVM).

To get a visual representation of our relevant subsets, we mapped it into a 2D space using the well-known JSE algorithm [37]. As seen in Fig. 9, the EUC metric preserves neighborhoods locally, showing that our dissimilarity representation codes the similarity between classes. Notwithstanding, globally, our features do not seem to embed well the information per

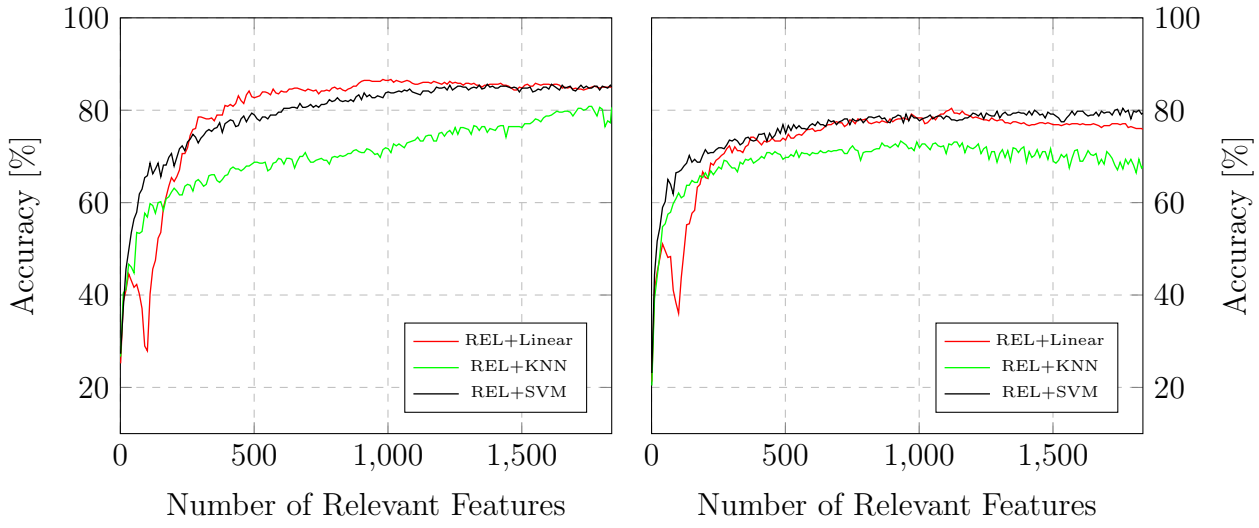

Figure 8: Classification performance against amount of relevant features for sampled PHGR DB. Left results corresponds to DTW-based representation using EUC metric, and right results using SKL metric.

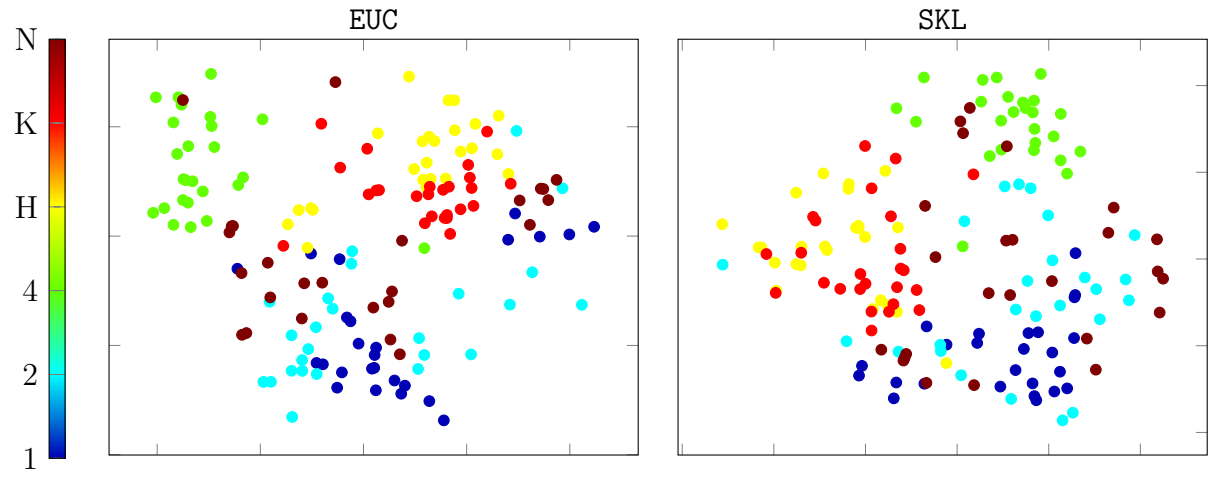

Figure 9: 2-D embeddings using JSE for both metrics used for sampled PHGR DB.

class. However, it is well known that the classification models discriminate against the selected attributes in the high dimensional space, showing that our approach preserves the discriminative ones from the database. Further, the SKL metric retains neither local nor global similarities, which restates that SKL as dissimilarity representation lacks in coding and preserving relevant information.

Table 8 shows the confusion matrix from the best classification results regarding the performance curves in Fig. 8. In particular, the best Linear classifier results are depicted for 851 relevant features according to the EUC metric. First of all, tackling the problem of SSLR from binary images is a difficult task since they present a low degree of homogeneity. Thus,

Table 8: Confusion matrix for the Linear Classifier using the 851 most relevant DTW-based features (EUC metric) for specific objective 2. Average±standard deviation are displayed in percentage.

	1	**2**	**4**	***H***	***K***	***N***
1	86 ± 14	7 ± 11	0 ± 0	0 ± 0	0 ± 0	6 ± 9
2	1 ± 4	85 ± 13	0 ± 0	1 ± 4	0 ± 0	13 ± 13
4	0 ± 0	5 ± 6	93 ± 6	0 ± 0	0 ± 0	3 ± 5
H	0 ± 0	0 ± 0	0 ± 0	79 ± 13	17 ± 12	4 ± 6
K	1 ± 4	0 ± 0	0 ± 0	14 ± 14	85 ± 14	0 ± 0
N	1 ± 4	14 ± 9	0 ± 0	1 ± 4	3 ± 5	81 ± 9

a system that discriminates correctly hands gestures needs to learn mostly each variation from the same sign. Nevertheless, the proposed methodology based in CCS represented by a DTW-based measurement resolves almost well the whole classes. However, between H-K and 2-N gestures tend to compromise the results since these such classes are similar in shape. Furthermore, Table 8 shows that information from small and large scales, e.g., $\xi_l = 3$ and $\xi_l = 20$, allows our system to learn details that between scales differs and determinate how the gesture is performed. Likewise, the EUC metric captures relevant information in the DTW-based representation, obtaining a high-performance per class. Thus, our methodology reveals information at different scales, which allows classifying hand gestures.

5.5 Summary

We develop a novel DTW-based representation approach to discriminate BI. Here, we develop a strategy to support SSLR. As we did to achieve the specific objective 1, we estimated CCS from BI of hand gestures. We did that computing a multi-step length feature estimation to reveal both local and global variations towards shape edges. Then, we computed a DTW-based distance to compare CCS of different lengths. We used an E FS approach to code consistent patterns regarding the DTW features from different step lengths. Hence, our FS stage allows us to prune redundant information from scales, revealing trends in the BI DB. Our proposal will enable us to code morphologic similarities to varying levels between BIs using data sequences, encouraging discrimination among static signs. Attained results on a public database prove that we can achieve an 85% classification performance on average. Besides, to the best of our knowledge, this methodology is the first attempt to apply dissimilarity-based representations to codify sequential data from BI devoted to SSLR. We used a simple set of features and powerful representation techniques to learn data-driven classification models.

6 Dimensionality Reduction for binary image classification and data visualization

6.1 Introduction

The last objective aims to develop a DR strategy for BI classification and visualization. That is a challenge since the State-Of-The-Art does not reveal any strategy to include such methods. This objective would complement the methodology we implemented so far since we will gather visual information from the previous stages involved in this IPS pipeline. In this section, the problems related to BI SC remains the same. We have to deal with HD information and low interpretability results [20–25, 30].

To develop a DR method to favor BI SC, we used the `CKA` algorithm [39]. This method learns a family of kernels from input data and try to *align* the outcome to a label kernel [48]. We selected this technique because, regarding other DR methods, these use class categories from each DB(See Sections 4.3.1, 5.3.1 and 6.3.1). Thus, we apply this algorithm over the set of selected features obtained from FS stages. In this section, we also explore the benefits of NE FS methods. We introduce LDR and NLDR methods to visualize the feature spaces get. In this section, we will detail our modified IPS pipeline.

6.2 Methods

6.2.1 Feature Extraction: Non-parametric Curvature-based Descriptors

Curvature Sequences To obtain curvature direction information, one can compute curvature using a signed version, which is a non-parametric form to get curvatures [49]:

$$\kappa = \frac{\left(\frac{dx_s}{dt}\right)\left(\frac{d^2y_s}{dt^2}\right) - \left(\frac{dy_s}{dt}\right)\left(\frac{d^2x_s}{dt^2}\right)}{\left(\left(\frac{dx_s}{dt}\right)^2 + \left(\frac{dy_s}{dt}\right)^2\right)^{3/2}} \tag{7}$$

Then one would have a set of CCS $\{\boldsymbol{\kappa}_{cn'}\}_{n'=1}^{N'}$ and a set of CS $\{\boldsymbol{\kappa}_{n'}\}_{n'=1}^{N'}$ from databases images. From here we are going to assume that a vector $\boldsymbol{x} = \{\boldsymbol{\kappa}_c, \boldsymbol{\kappa}\}$

6.2.2 Feature Representation: BoCF-based representation of binary images

Let $S(t) = (x_s(t), y_s(t))\ t \in [0, 1]$ represent a shape, which it is described by an outer contour. `BoCF` [20] builds a shape representation vector $\boldsymbol{x} \in \mathbb{R}^+\,[0, 1]$ from $S(t)$ following three stages:

FE, codebook construction, and vectorial quantization. To extract features, it is applied a technique named as DCE [30], which allows to get a simplified polygon from S where their T vertices are $\boldsymbol{v} = (v_1, v_2, ..., v_T)$. They are computed according to a threshold τ. Once each vertex is obtained, a set of contour fragments (CF) $\mathcal{C}(S)$ is build between each pair (v_i, v_j) following segments. Thus, let c_{ij} be a CF between v_i and v_j, not necessarily adjacent, which follows $\mathcal{C}(S) = \{c_{ij} = (v_i, v_j), i \neq j, i, j \in [1, ..., T]\}$. This procedure gives, for each CF, a context shape descriptor $\boldsymbol{\theta}_{ij} \in \mathbb{R}^{d\times 1}$ where d is c_{ij} dimension. Then, P reference points are sampled from c_{ij} between v_i and v_j, computing a H histograms, giving a descriptor which concatenates P histograms of shape context. To build the shape codebook $\boldsymbol{B}$, each $\boldsymbol{f}_{ij}$ is mapped into a new space $\mathbb{B}$, spanned by $\boldsymbol{B}$, whose representation relies on shape codes $\boldsymbol{w}_{ij}$. To get $\boldsymbol{B}$, a K-means clustering is employed, building a $\boldsymbol{B} = [\boldsymbol{b}_1, ..., \boldsymbol{b}_M] \in \mathbb{R}^{d\times G}$, where each column G is a cluster centroid, or even, a prototype which describes the whole new space. To get vectorial quantization Local-constraint Linear Coding is used to represent $\boldsymbol{\theta}_{ij}$ in the $\mathbb{B}$ space. This method uses K nearest neighbors in $\mathbb{B}$ as local basis for $\boldsymbol{\theta}_{ij}$ to build a local coordinates system. The K nearest neighbors of $\boldsymbol{\theta}_{ij}$ are given by $\boldsymbol{B}_{\boldsymbol{\pi}_{ij}} \in \mathbb{R}^{d\times k}$ where $\boldsymbol{\pi}_{ij} = \{\pi_{ij}^1, ..., \pi_{ij}^k\}$ represents the set of k nearest neighbors in $\boldsymbol{B}$. Hence, to obtain $\boldsymbol{w}_{\boldsymbol{\pi}_{ij}} \in \mathbb{R}^{k\times 1}$ the minimization problem

$$\min_{\boldsymbol{w}_{\boldsymbol{\pi}_{ij}}} \|\boldsymbol{\theta}_{ij} - \boldsymbol{B}_{\boldsymbol{\pi}_{ij}} \boldsymbol{w}_{\boldsymbol{\pi}_{ij}}\| \text{ subject to } \mathbf{1}^\top \boldsymbol{w}_{\boldsymbol{\pi}_{ij}} = 1, \tag{8}$$

is solved. Since BoCF are developed, an additional stage is introduced known as Spatial Pryamid Matching. This method adds spatial disposition information to the representation. In this approach, each shape $S(t)$ is divided in l' regions of $2^{l'-1} \times 2^{l'-1}$, where $l' \in [1, ..., L], L \in \mathbb{Z}$. Thus, we obtained $R_{r'}, r' \in [1, ..., \nu], \nu = \sum_{\forall l'} 2^{l'-1} \times 2^{l'-1}$, subregions were max pooling is developed. Let $\boldsymbol{w}_u$ denote a CF codification at u shape position, the max-pooling $\Gamma_r = \theta(S, R) = \max(\boldsymbol{w}_u | u \in R'_r)$. Then for each CF, the max value of the codes set is taken to do the shape representation. Finally, one have a matrix $\boldsymbol{X}_s \in \mathbb{R}^{N'\times Q}$, where N' is the number of images of the DB, $Q = r' \cdot G$ are the BoCF descriptors, and each vector concatenates the features per region $\boldsymbol{x}_{s_n} = \left[x_{s_1}^\top, ..., x_{s_{r'}}^\top\right]^\top$.

6.2.3 Feature Selection: LASSO applied to curvature-based features

To reduce the IPS complexity, we proposed to apply FS to prune redundant features. However, we implement an E FS method, which is a brute force technique to get the ideal set of features that preserves the IPS performance to discriminate BI. Yet, there are plenty of methods that try to find the optimum set of features trough NE search. Here, we introduce LASSO, which is a sparse regularization method [50]. This technique is used as a supervised feature selection

scheme. In this approach, it is solved the l_1 problem

$$\begin{aligned} \underset{\boldsymbol{M}}{\text{minimize}} \quad & \left(\frac{\|\boldsymbol{y} - \boldsymbol{M}\beta\|_2^2}{n'} \right) \\ \text{subject to} \quad & \sum_{j=1}^{k} \|\beta\|_1 < \epsilon. \end{aligned} \tag{9}$$

t is the coefficients sum upper bound. The previous optimization problem is equivalent to estimate the regularization parameters as

$$\hat{\beta}(\lambda) = \arg\min_{\boldsymbol{\beta}} \quad \left(\frac{\|\boldsymbol{y} - \boldsymbol{M}\beta\|_2^2}{n} + \lambda \|\beta\|_1 \right) \tag{10}$$

If $\epsilon \to \infty$, $\lambda = 0$ and the regularization problem is identical to the ordinary least square method. Instead, if $\epsilon \to 0$, $\lambda \to \infty$, the whole coefficients are reduced to zero. Its key advantage is that the regularization coefficients are binary flags that choose which feature to preserve and which one should be discard [51].

6.2.4 Dimensionality Reduction: Centered Kernel Alignment to support binary image classification

Kernel Alignment was first introduced by authors in [48]. This method is to measure the similarity between kernels and matrices. Authors in [39] proposed to center this measure to enhance its performance. Its key advantage is that the algorithm relieves us from specifying features. Thus, it is a supervised DR method. Let $\boldsymbol{K}_{XX} \in \mathbb{R}^{N' \times N'}$ be the kernel over the input data matrix, and let $\boldsymbol{K}_{yy} \in \mathbb{R}^{N' \times N'}$ be the kernel matrix from the $\boldsymbol{y}$ label vector. Both matrices follows that $\|\boldsymbol{K}_{XX_c}\|_F \neq 0$ and $\|\boldsymbol{K}_{yy_c}\|_F \neq 0$. The alignment between the two kernel matrices is deemed as

$$\hat{\rho}(\boldsymbol{K}_{XX}, \boldsymbol{K}_{yy}) = \frac{\langle \boldsymbol{K}_{XX_c}, \boldsymbol{K}_{yy_c} \rangle}{\|\boldsymbol{K}_{XX_c}\|_F \|\boldsymbol{K}_{yy_c}\|_F} \tag{11}$$

where $\boldsymbol{K}_{XX_c}$, $\boldsymbol{K}_{yy_c}$ are centered kernel matrices which

$$\boldsymbol{K}_c = \left[\boldsymbol{I} - \frac{\boldsymbol{1}\boldsymbol{1}^\top}{N'} \right] \boldsymbol{K} \left[\boldsymbol{I} - \frac{\boldsymbol{1}\boldsymbol{1}^\top}{N'} \right] \tag{12}$$

The alignment between both matrices is a similarity measure, because $\hat{\rho}(\boldsymbol{K}_{XX}, \boldsymbol{K}_{yy}) \in [-1, 1]$, but since the Frobenius product between two positive semi-definite (PSD) matrices is non-negative, this measure follows $\hat{\rho}(\boldsymbol{K}_{XX}, \boldsymbol{K}_{yy}) \geq 0$. However, we introduce `CKA` as a DR method. thus, we defined the kernel matrix $\boldsymbol{K}_u$ as

$$\boldsymbol{K}_u \propto \sum_{q'=1}^{\bar{p}} \hat{\rho}(\boldsymbol{K}_{XX_{q'}}, \boldsymbol{K}_{yy}) \boldsymbol{K}_{XX_{q'}} = \frac{1}{\|\boldsymbol{K}_{yy}\|_F} \sum_{q'=1}^{\bar{p}} \frac{\langle \boldsymbol{K}_{XX_{q'}}, \boldsymbol{K}_{yy} \rangle}{\|\boldsymbol{K}_{XX_{q'}}\|_F} \boldsymbol{K}_{XX_{q'}} \tag{13}$$

where $\boldsymbol{K}_{XX_{q'}}$ is a combination of Q based kernels where $q' \in [1, Q]$. The $\boldsymbol{K}_u$ matrix is non-centered, thus $\boldsymbol{K}_{u_c} = \boldsymbol{U}_{N'} \boldsymbol{K}_u \boldsymbol{U}_{N'}$, where $\boldsymbol{U}_{N'} = \boldsymbol{I} - \boldsymbol{1}\boldsymbol{1}^\top / N'$. Therefore

$$\boldsymbol{K}_{u_c} = \sum_{q'=1}^{\bar{p}} u_{q'} \boldsymbol{K}_{XX_{q'_c}} \tag{14}$$

the $u_{q'}$ are values which guarantees that $\boldsymbol{K}_{XX_{q'_c}}$ is PSD. Let $\boldsymbol{a} = \big(\langle \boldsymbol{K}_{1_c}, \boldsymbol{y}\boldsymbol{y}^\top \rangle, ..., \langle \boldsymbol{K}_{q'_c}, \boldsymbol{y}\boldsymbol{y}^\top \rangle\big)$. These vectors allows to define a matrix which comprises the input kernel information associate to the image DB:

$$\boldsymbol{W}_{q',l'} = \langle \boldsymbol{K}_{XX_{q'_c}}, \boldsymbol{K}_{XX_{l'_c}} \rangle_F \tag{15}$$

and $q', l' \in [1, Q']$. $\boldsymbol{W}$ is a PSD matrix since for any input vector $\boldsymbol{X} = (x_1, ..., x_{q'})^\top$. Remember that Q' is the set of selected features from FS stages. Then, $\boldsymbol{W}$ is a projection matrix which defines a new l' dimension that follows $l' < Q'$. Hence, `CKA` works like a DR method. Our proposal applies this technique to learn a set of kernel matrices which allows us to map our data to an RKHS. There, we get a projection matrix that prunes redundant features that might be filtered during the FS stages.

6.3 Experimental Set-up

6.3.1 Databases

To test our method, we used four DBs: `Kimia 99 Shape` DB, `MPEG7` DB, `PHGR` DB, and introduced the more challenging DB in the State-Of-The-Art the `Animals Dataset` [4] - (`Animals` DB). The `Animals` DB have 20 classes. Each category has 100 samples. We described in detail the other DBs in the previous Section 4.3.1 and Section 5.3.1. Every DB has its own properties. `Kimia 99 Shape` DB, `Animals` DB and `MPEG7` DB are heterogeneous, since each of them presents intra-class and inter-class non-uniformity. However, `PHGR` DB is homogeneous because their classes are uniform, being more challenging. Fig. 10 shows samples from the `Animals` DB.

6.3.2 Quality Assessment

To assess the proposed ensemble, we get `CCS` as we exposed in Section 4.2.1, which corresponds to a set of parametric curvatures. Thus, a curvature depends on the ξ_l parameter and how many of them (η) we computed. In fact, we got `CCS` as many images the DB has, and we computed every `CCS` at 18 different scales. Still, we introduced here `CS` (See Section 6.2.1),

[4] https://sites.google.com/site/xiangbai/animaldataset

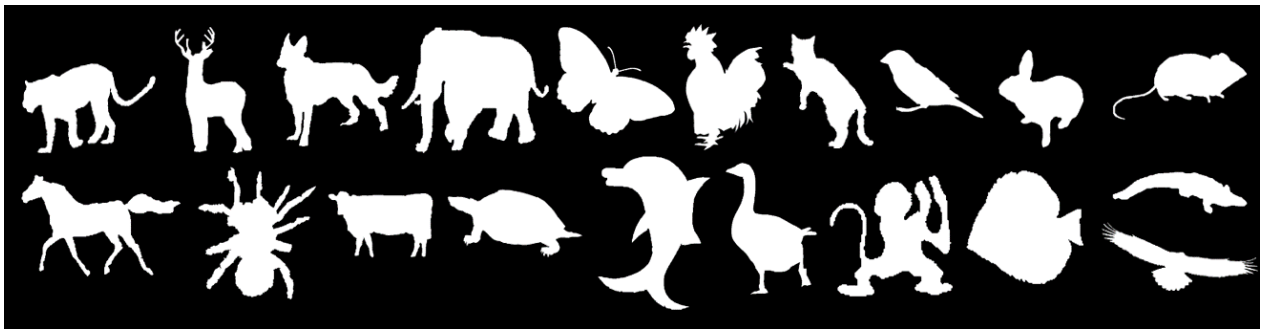

Figure 10: Samples from `Animals` DB used in this work.

which is a non-parametric version of `CCS`. This FE method allowed us to know whether to get scale information is useful. Hence, we compute `CS` sets proportional to the number of samples of each DB. We note that each `CCS` and `CS` length is different because each one depends on the contour trajectory extent.

To represent `CCS` and the `CS` we used a `DTW`-based approach to encode similarity/dissimilarity patterns. We discard the `KAF`-based methods because these techniques introduces more complexity into the BI SC task. Thus, we compute distance matrices between `CCS` sets `CS` sets. The major difference between them depends on `CCS` codes scale information in a parametric manner while CS does not because is a non-parametric approach (See Section 6.2.1). Hence, we got a set of $\boldsymbol{D}^l$ for `CCS` and a unique $\boldsymbol{D}$ for `CS`. We introduced here the `BoCF` FE scheme. This method extract curvature and then computes a shape codebook ($\boldsymbol{B}$) which gather descriptors from high and low scales of segments of curvatures of the image. At the end, it is described the DB by a matrix $\boldsymbol{X}_s$ which comprises $\boldsymbol{x}_{s_{n'}}$ vectors of each region derived by pyramid pooling [20]. Then, in this stage, we got three types of matrices representing `DTW-CCS`, `DTW-CS`, and `BoCF` approaches for every DB used in this methodology. Fig. 11 shows `CCS` and `CS` examples from every DB involved in this work.

To prune redundant information, we used E and NE FS methods to assess features according to relevance. The E method used was ReliefF, and we implemented it as the Section 5.2.3 exposes. However, we explored another approaching trough the NE scheme named LASSO (See Section 6.2.3). The last method requires creating a grid of penalization parameters λ and threshold value to filter the features. We used 20 unique values for λ, and we changed the threshold from $1{\times}10^{-6}$ to $1{\times}10^{-1}$ at an equally distant range. Hence, as we did in Section 5, we used this filtered feature space $\boldsymbol{F}'$ to classify shapes.

To assess our methodology, we implement three classifiers: Linear, Logistic Regression, and Linear SVM. We choose those because we are trying to filter the attribute space, and it is possible do not require such a complex decision region to discriminate BI. To tune the parameters of every classifier, we proceeded like this: we used the stochastic gradient descent

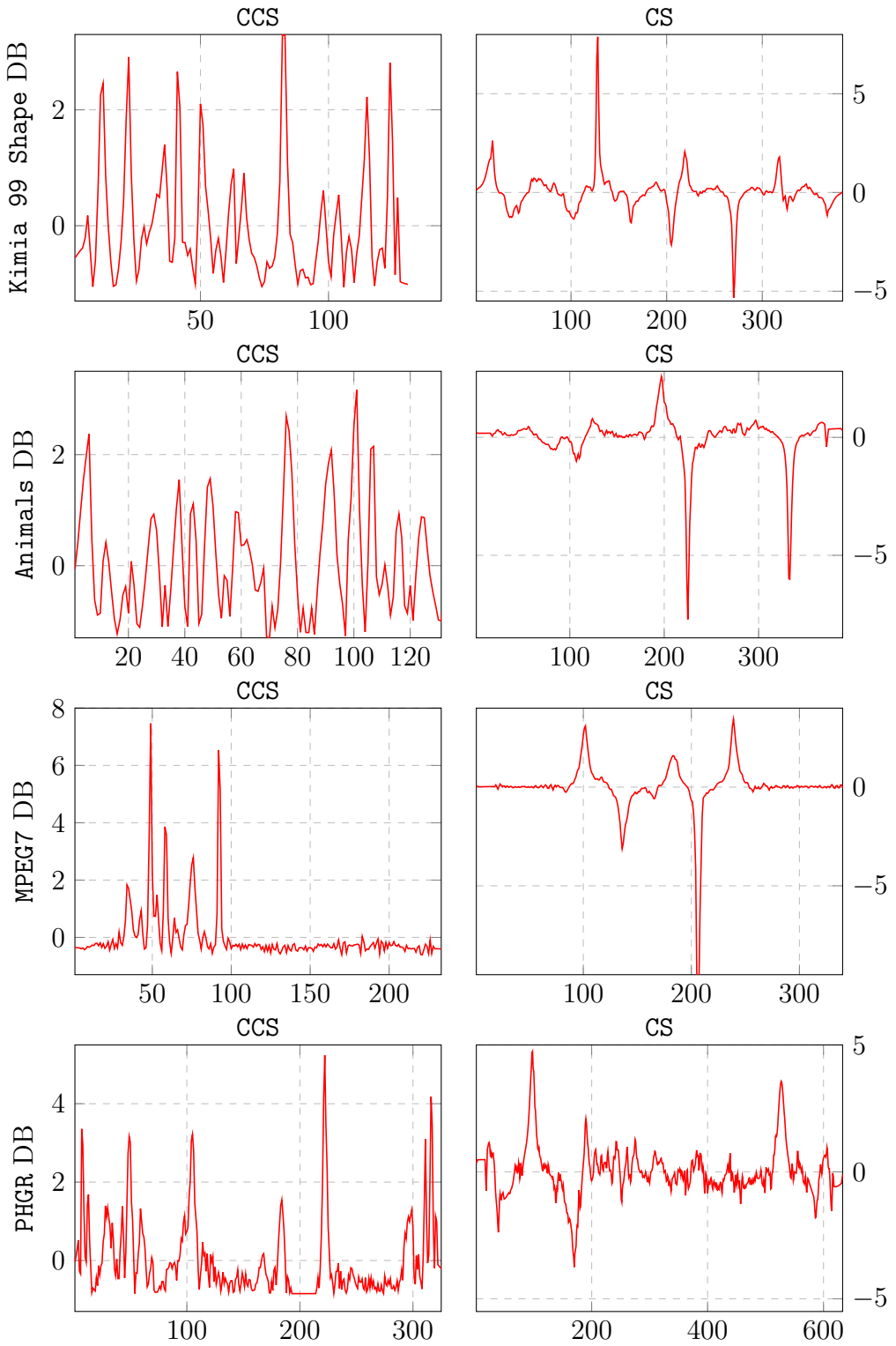

Figure 11: CCS and CS samples from every DB used in this work

implementation of Linear classifier, varying penalization trough `l-1` norm, `l-2` norm, and `elasticnet`. We use the penalization range $0.01, 0.1, 1, 10$ to tune Logistic Regression, and the Linear SVM requires setting the regularization parameter varying it between the grid $0.1, 1, 10, 100, 1000$. To train these classifiers and selected the optimum number of features, we used a 10-fold nested cross-validation scheme, using Hold-Out approach $(50 - 50)\%$ training-testing as the State-Of-The-Art suggest [20].

To prune redundant information filtered during FS and classification stages, we applied an appearance-based feature filtering. That means we keep attributes selected at least 60% at the 10-fold runs. Thus, we got a reduced feature space. To reveal deeper patterns, we implement an NLDR technique named `CKA`, used as a reduction technique due to computing a projection matrix [39]. To apply this technique, we used a mini-batch version. We fixed the size of the batch as equal to or greater than the number of classes of each DB. Hence the sizes chosen were 30 for `Kimia 99 Shape` DB, `Animals` DB, and `PHGR` DB; and 70 for `MPEG7` DB. `CKA` method as an NLDR technique preserves the variance since it initializes using `PCA`. Therefore, we fixed the variance value at 0.95, trying to preserve as much variance as the reduced feature space holds. Once we run `CKA` we got a more reduced space, which we introduced again into the classification stage. We assessed the variance preserve into this new representation. The parameters for this new classification phase remain the same as we exposed.

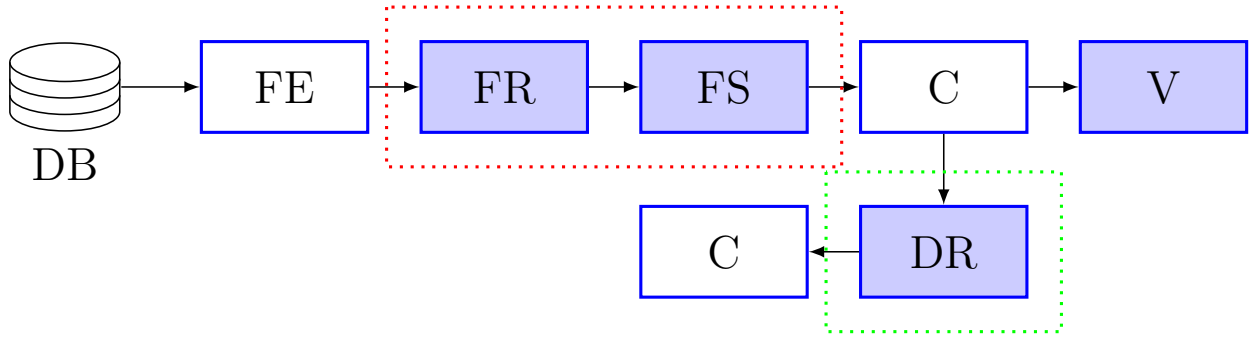

Figure 12: Dimensionality Reduction for BI classification and data visualization flowchart. Blue filled rectangles represents the stages we add and/or modify to enhance the BI SC performance and interpretability. Green rectangle covers the DR stage where we applied `CKA` to discard non-informative features.

To visualize information, we also used DR. We implemented NLDR methods to get insights about the DBs. The techniques chosen were `tSNE`, `SNE`, `JSE`, `NeRV`, and `CKA`. The LDR approach used was `PCA`. Each DR method mapped the reduced space into a 2-dimensional space for sake of visualization. However, `tSNE`, `SNE`, `JSE`, and `NeRV`, need to set the size of the neighborhood or perplexity. We set that value as the number of samples per class divided by two [37]. Thus, for `Kimia 99 Shape` and `MPEG7` DB this rate was 10, for `Animals` DB was

50, and PHGR DB was 12. To visualize groups using CKA and PCA both methods demands to fixed the amount of variance preservation. Thus, we set this value in 95%, plotting the first two dimensions, because according to the theory those preserves the highest feature space variance [39]. Fig. 12 summarizes the ensemble of proposed methodologies to achieve the goal of the specific objective 3.

6.3.3 Method Comparison

As a baseline, we used different benchmarks based on skeleton and contour-like methods. Here we introduce the techniques that we used to compare our methodology: the Inner Distances Shape Context (IDSC) [11], Skeleton Paths, Contour Segements, and Integrated Skeleton Paths and Contour Segments (ICS) [12], Kernel-edit Distance (KeD) [15], BoCF [20], Line Segment Statistics (LSS) [18], BoCF and Bag of Skeleton Paths(BoCF+BoSP) [21], Bioinformatics [19], Contextual BOW model (CBOW model) [25], Bag of Skeleton-associated Contour Parts (BoSCP) [22], BoCF and BoSCP using a Learning Pooling function (BoCF-LP and BoSCP-LP) [24], RNN [29], Curvature Bag of Words (CBoW) [30], and Enlacement and Interlacement Shape Descriptor (EID) [31].

6.4 Results and Discussion

This section covers the results of the proposed methodology. Since we introduce new FS, DR, and visualization methods to our IPS pipeline, we divide this part into two categories: FS results and DR results.

6.4.1 Feature Selection: Kimia 99 Shape Database

Table 9 presents results in State-Of-The-Art against our scheme. One of our key advantages regarding the other methodologies is that we develop a complete set of experiments, as we exposed in Section 6.3. That means we have a more robust experimental setup since we are reporting mean accuracy and deviation. Farther, our method is better than KeD since we do not have to compute a kernel matrix. That means we do not increase the IPS complexity. Thus, we get a simpler FR. Besides, our proposal overcomes LSS because it can deal with BI occlusions showing that our method is more potent.

Regarding the interpretation of the results, we get a feature space which we can understand as a product of scale concatenation from BoCF features. Hence, our technique surpasses RNN since we can tack the regions where our features came from, while RNN produces a set on

Table 9: `Kimia 99-Shape` DB: our results against State-Of-The-Art benchmarks

Algorithm	**Number of Features**	**Classifier**	**Accuracy**%
`KeD` [15]	99	Gaussian SVM	$98.93 \pm -$
`LSS` [18]	34	LDA	$96.96 \pm -$
`RNN` [29]	40	LDA	$97.98 \pm -$
`EID` [31]	32	Gaussian SVM	$97.00 \pm -$
Ours	**900**	**Logistic Regression**	$\mathbf{99.80 \pm 0.60}$

non-linear combinations of features, leading to see the method as a black-box procedure. One of the main problems of `EID` is its loss of relevancy when the number of classes starts to grow. That forces to combine these kinds of features with others. Notwithstanding, our proposal can deal with this problem, as we are going to see in the subsequent sections. Yet, our method fails regarding the number of features that it gets. But, more than a problem is an advantage as we are going to explain with the results from Table 10.

Table 10 shows the complete set of experiments we develop over the `Kimia 99 Shape DB`. As we exposed in Section 6.2, we have multiple contour-based features extracted from each image. As a matter of fact for this DB, representations like `DTW-CS`, `DTW-CCS` and `BoCF`, shows that this kind of representation resolves the recognition of binary images based on its shape. However, a prototype-like representation based on `BoCF` (PR-`BoCF`), does not help our recognition system. This might be `BoCF` features captures information of different scales ought to pyramid pooling [20]. Then, when PR is computed, this information could be lost, yielding to a poor representation of patterns that allows identifying binary shapes.

Respect the feature selection scheme, Table 10 shows its strength since, for each representation proposed, the feature space turns out in a smaller one, not compromising performance. Further, this DB has fewer samples than other DBs treated in this work, which could yield over-fitting. Then, the FS techniques proposed in this work support generalization for the whole scheme of recognition, since ReliefF and LASSO methods select a quality set of features. In fact, performance over `DTW-CS`, `DTW-CCS`, and `BoCF` present accuracy results that resolve almost every class of the DB.

Table 9 and Table 10 show the strongest result is the `BoCF`-ReliefF. Under the `BoCF` scheme, we extracted 31500 features, and once we applied ReliefF over this feature space, 900 features persisted. That means under this representation, near 3% of the dataset describes each class. But, we do not select arbitrary these features. Fig. 13 denotes `BoCF`, a scheme based on scales ought to pyramid pooling [20], the regions where a vast amount of information is capture are those related to high scales (see Section 6.2.2). Even when we filtered these attributes according to their percentage of appearances, the following histogram keeps the same tendency of the non-filtered one. Thus, our method illustrates that the original `BoCF`

Table 10: `Kimia 99-Shape` DB results for `DTW-CS`, `DTW-CCS` and `BoCF` representations and subsequent FS

Algorithm	Number of Features	Classifier	Accuracy%
		Linear	84.80 ± 5.60
`DTW-CS`	99	Logistic Regression	91.00 ± 4.58
		Linear SVM	92.00 ± 5.29
	82	Linear	85.40 ± 4.98
`DTW-CS`-ReliefF	69	Logistic Regression	90.39 ± 3.07
	69	Linear SVM	90.40 ± 3.07
	19	Linear	83.80 ± 4.42
`DTW-CS`-LASSO	21	Logistic Regression	86.80 ± 4.02
	20	Linear SVM	91.19 ± 5.31
		Linear	82.40 ± 5.12
`DTW-CCS`	1782	Logistic Regression	88.80 ± 3.37
		Linear SVM	89.40 ± 3.58
	866	Linear	90.99 ± 3.49
`DTW-CCS`-ReliefF	765	Logistic Regression	91.00 ± 2.86
	765	Linear SVM	91.59 ± 2.94
	81	Linear	78.40 ± 14.28
`DTW-CCS`-LASSO	85	Logistic Regression	85.00 ± 9.64
	75	Linear SVM	83.40 ± 10.43
		Linear	83.00 ± 7.65
`BoCF`	31500	Logistic Regression	83.00 ± 7.33
		Linear SVM	83.00 ± 7.05
	825	Linear	100.00 ± 0.00
`BoCF`-ReliefF	**900**	**Logistic Regression**	$\mathbf{99.80 \pm 0.60}$
	750	Linear SVM	100.00 ± 0.00
	42	Linear	93.20 ± 7.28
`BoCF`-LASSO	42	Logistic Regression	93.20 ± 6.82
	42	Linear SVM	94.20 ± 6.41
		Linear	13.60 ± 2.94
PR-`BoCF`	99	Logistic Regression	14.60 ± 4.56
		Linear SVM	16.80 ± 5.23
	16	Linear	$13,80 \pm 4,51$
PR-`BoCF`-ReliefF	20	Logistic Regression	$11,80 \pm 2,89$
	20	Linear SVM	$15,60 \pm 4,88$
	17	Linear	$12,80 \pm 1,33$
PR-`BoCF`-LASSO	**15**	**Logistic Regression**	$\mathbf{11,80 \pm 1,66}$
	20	Linear SVM	$16,80 \pm 6,52$

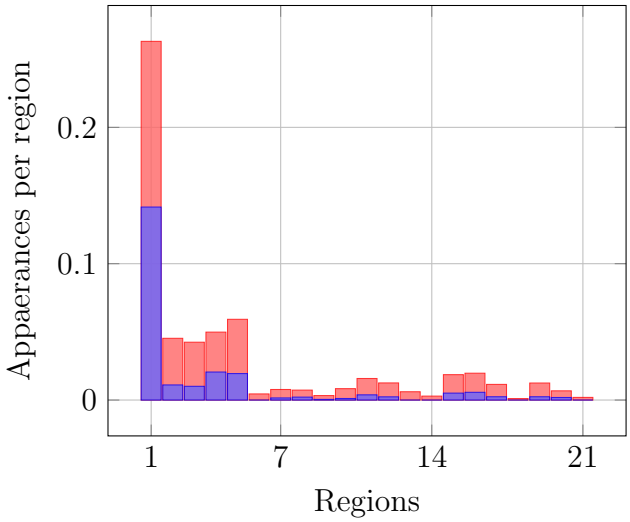

Figure 13: Histogram of selected feature by region for `Kimia 99 Shape DB`

representation is redundant, allowing us to prune those features that do not give any extra information on the binary shapes.

Our proposed methodology has an overall competitive performance regarding the other State-Of-The-Art techniques for `Kimia 99 Shape DB`. Besides, we compared our method concerning the proposed stages that they implemented. Indeed, `LSS` and `RNN` methodologies implement an FS and classification stage, which means they did not map the feature space to another. Thus, we showed that the performance could increase if we add an FR stage to the pipeline. Also, regarding `KeD` and `EID` methods, we derived a technique that can deal with the lack of interpretability and scalability of those methods improving the IPS performance.

6.4.2 Feature Selection: Animals Database

Table 11 presents our results against other State-Of-The-Art methods. In this DB, got results are competitive against other techniques. As we mentioned in Section 6.3, we used an FR method that involves an FE stage that can overcome the shape-related problems like non-closed boundaries. This ability exposes that our method is better than Skeleton Paths, Contour Segments, and `ICS`. As a result, our method overcomes the performance of the methodologies, as mentioned earlier. Regarding the Shape Vocabulary method, we reach a comparable result. However, our technique includes scale information, while that method does not. The Bioinformatics approach shows high performance since the DB exposes high inter-class variability. Our proposal can deal with that too, but the number of features is greater, showing that Bioinformatics tools overcome our method. Concerning `BoCF`-based representations [20–25], our methodology surpasses their results since we get a lower set of

features from BI. This result ensures that our method can avoid over-fitting introduced because the number of features of these techniques is greater. Thus, we discard those redundant features, ensuring that we can reveal the relevant patterns trough the scale-based features.

Table 11: `Animals` DB: our results against State-Of-The-Art benchmarks

Algorithm	**Number of Features**	**Classifier**	**Accuracy**%
Skeleton Paths [12]			$67.90 \pm -$
Contour Segments [12]	Subject to FE method	AdaBoost	$71.70 \pm -$
`ICS` [12]			$78.40 \pm -$
`BoCF` [20]	31500	Linear SVM	83.40 ± 1.40
Shape Vocabulary [17]	300	KNN	84.30 ± 1.01
`BoCF+BoSP` [21]	52500	Linear SVM	85.50 ± 0.88
Bioinformatics [19]	2000	KNN	$83.70 \pm -$
`CBOW` model [25]	36275	Gaussian SVM	$86.00 \pm -$
`BoSCP` [22]	52500	Linear SVM	89.04 ± 0.95
`BoCF-LP` [23]	31500	Max pooling function	86.30 ± 0.2
`BoCF-LP` [24]	31500	Pooling function	86.30 ± 0.2
`BoSCP-LP` [24]	31500	Pooling function	89.77 ± 0.65
Ours	**11550**	**Logistic Regression**	$\mathbf{85,17 \pm 0,95}$

Table 12 presents the complete set of results using `DTW-CS`, `DTW-CCS`, and `BoCF` schemes. Here, the `Animals` DB has 2000 samples, and each class has 100 images. Thus, this DB is more complex due to the number of instances and the intra-class heterogeneity. That induces the system to learn more intricate decision regions, inducing more false predictions. The descriptors got from `DTW-CS` and `DTW-CCS` do not resolve well the DB. Then, a pure curvature-based portrayal, no matter if it is parametric or non-parametric, shows poor performance. Hence, `BoCF` and PR-`BoCF` schemes show comparable results to the other benchmarks. The PR-`BoCF` technique using a Linear SVM performs well on this dataset. PR-`BoCF` itself reduces the number of features, due to proceed from a similarity/dissimilarity based matrix. However, when we applied FS, performance its reduce. Then, ReliefF applied over the `BoCF` space turns out to be more reliable, since a smaller dataset preserves as much information as the whole dataset from the `BoCF` complete space.

Table 11 and Table 12 summarizes the work done over `Animals` DB. `BoCF`-ReliefF ensemble shows that performance in this DB is great since it preserves non-redundant information. When we computed the `BoCF` features, we got 31500 descriptors, but when we applied ReliefF, it reduced this space to 11550. That quantity of attributes corresponds to a 37% feature space reduction, inducing to recognize each class. However, this percentage decays since we filter

Table 12: `Animals` DB results for `DTW-CS`, `DTW-CCS` and `BoCF` representations and subsequent FS

Algorithm	Number of Features	Classifier	Accuracy%
		Linear	41.83 ± 1.58
`DTW-CS`	2000	Logistic Regression	47.04 ± 1.58
		Linear SVM	44.12 ± 1.43
	1680	Linear	$41,82 \pm 1,55$
`DTW-CS`-ReliefF	1800	Logistic Regression	$46,77 \pm 1,69$
	1680	Linear SVM	$44,30 \pm 1,11$
	52	Linear	$32,28 \pm 3,11$
`DTW-CS`-LASSO	51	Logistic Regression	$38,62 \pm 2,28$
	49	Linear SVM	$36,71 \pm 2,27$
		Linear	$40,69 \pm 1,12$
`DTW-CCS`	36000	Logistic Regression	$44,59 \pm 1,33$
		Linear SVM	$42,42 \pm 1,38$
	25900	Linear	$39,92 \pm 1,16$
`DTW-CCS`-ReliefF	22900	Logistic Regression	$44,76 \pm 1,49$
	27000	Linear SVM	$41,66 \pm 0,89$
	88	Linear	$28,58 \pm 2,43$
`DTW-CCS`-LASSO	89	Logistic Regression	$34,53 \pm 2,16$
	77	**Linear SVM**	$\mathbf{27,46 \pm 1,10}$
		Linear	$80,64 \pm 0,62$
`BoCF`	31500	Logistic Regression	$84,62 \pm 0,83$
		Linear SVM	$82,38 \pm 0,83$
	17775	Linear	$80,34 \pm 0,90$
`BoCF`-ReliefF	**11550**	**Logistic Regression**	$\mathbf{85,17 \pm 0,95}$
	23250	Linear SVM	$81,84 \pm 0,62$
	622	Linear	$72,26 \pm 3,79$
`BoCF`-LASSO	626	Logistic Regression	$75,11 \pm 4,68$
	647	Linear SVM	$65,47 \pm 5,73$
		Linear	$71,99 \pm 0,71$
PR-`BoCF`	2000	Logistic Regression	$77,69 \pm 0,91$
		Linear SVM	$81,76 \pm 0,81$
	560	Linear	$72,21 \pm 1,06$
PR-`BoCF`-ReliefF	800	Logistic Regression	$78,36 \pm 1,40$
	800	Linear SVM	$76,85 \pm 1,35$
	480	Linear	$72,25 \pm 0,87$
PR-`BoCF`-LASSO	479	Logistic Regression	$78,67 \pm 1,28$
	479	Linear SVM	$79,25 \pm 1,53$

this feature space, taking into account each attribute's importance. Fig. 14 shows how many proportions of features per region prevail. As we exposed in Section 6.4.1, the trend is clear: the high scale information is more relevant than the low scale one. Still, in this DB, low scales are persistent ought to the DB extension (remind that there are 2000 images).

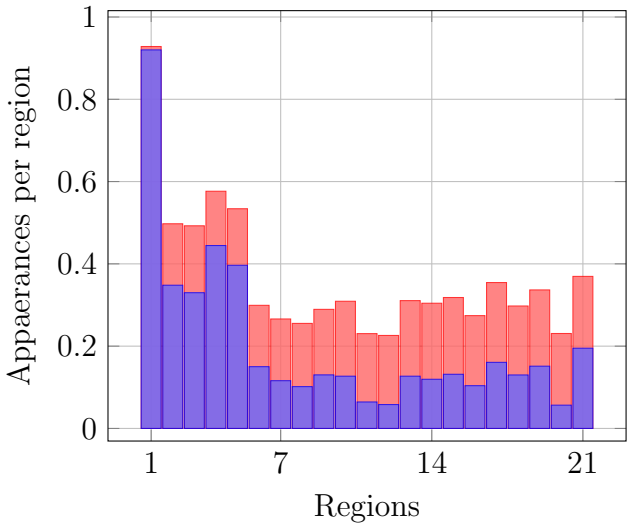

Figure 14: Histogram of selected feature by region for `Animals DB`

The results of this section show that FR stage implementation is essential. This phase ensures that the extracted features can be interpreted in a derived feature space, revealing the critical differences between the classes. However, as Table 11 shows, the FR method increases the complexity regarding other State-Of-The-Art techniques, showing that FS is not enough to prune the redundant information. Thus, we should include a subsequent stage that filters those features that might be present in this derived set of selected attributes.

6.4.3 Feature Selection: MPEG7 Database

Table 13 shows our results versus other benchmarks from State-Of-The-Art. As with `Animals` DB, our results are competitive regarding other methods. As we mentioned in Section 6.4.2, our proposal can deal with non-closed contours. Thus, for this DB, we get a greater performance than Skeleton Paths and Contour Segments, but `ICS` overcomes our results. The intra-class homogeneity of `MPEG7` DB might explain that behavior. Regarding `KeD` and Bioinformatics approaches we get lower results than them because `KeD` metric and `Bioinformatics` features reveal relevant shape changes. Concerning `BoCF`-based schemes [20–24, 30] these methods generate a higher space of features, leading to possible overfitting. However, our technique reaches good results, since the number of features gets

far lower than those BoCF-based techniques. Hence, the method generalization improves baseline results.

Table 13: MPEG-7 DB: our results against State-Of-The-Art benchmarks

Algorithm	**Number of Features**	**Classifier**	**Accuracy**%
Skeleton Paths [12]			$86.70 \pm -$
Contour Segments [12]	Subject to FE method	AdaBoost	$90.90 \pm -$
ICS [12]			$96.60 \pm -$
KeD [15]	1400	Gaussian SVM	$96.60 \pm -$
BoCF [20]	31500	Gaussian SVM	97.16 ± 0.79
BoCF+BoSP [21]	52500	Linear SVM	98.35 ± 0.63
Bioinformatics [19]	1400	KNN	$96.10 \pm -$
BoSCP [22]	52500	Linear SVM	98.41 ± 0.63
BoCF-LP [23]	31500	Max pooling function	98.22 ± 0.20
BoCF-LP [24]	31500	Pooling function	98.22 ± 0.20
BoSCP-LP [24]	31500	Pooling function	98.72 ± 0.42
CBoW [30]	31500	Linear SVM	$98.21 \pm -$
Ours	**4200**	**Logistic Regression**	$\mathbf{94,41 \pm 0,73}$

Table 14 presents the whole set of results for different representations of our work: DTW-CS, DTW-CCS, BoCF, and PR-BoCF. Each representation has advantages and disadvantages in this dataset. First of all, MPEG7 DB has inter-class heterogeneity. DTW-CS and DTW-CCS show a performance that is not comparable to that from Table 13. Despite the application of FS over these spaces, the performance remains almost the same as the complete version. Then, curvature-based methods do not resolve well the DB, as the other benchmark techniques do. Regarding BoCF-based features, classification results are comparable. Especially when we applied LASSO FS, the Linear SVM discriminates almost the entire database. However, in terms of performance, the ReliefF method applied over MPEG7 DB reaches the benchmarks' previous outcomes, reducing the number of attributes used to describe the DB.

Summarized results of the proposed method in Table 13 and Table 14 show that our work reveals relevant information through FS in this DB. As well as previous DBs, BoCF representation comprises a space of 31500 attributes, which could lead to over-fitting. Once we run the ReliefF method over this dataset, Q' =4200. That amount of descriptors corresponds to near 13% of the original space. Thus, facts like the inter-class heterogeneity and per class number of samples could positively impact performance. Yet, when we filtered features according to appearance criteria, a lower amount of attributes can describe the entire DB. Fig. 15 notice as in the previous results BoCF introduces redundant information once it computes another scales. Our approach shows high scales such as $r' = 1, 2, 3, 4, 5$ from

Table 14: MPEG-7 DB results for DTW-CS, DTW-CCS and BoCF representations and subsequent FS

Algorithm	Number of Features	Classifier	Accuracy%
		Linear	$72,54 \pm 2,25$
DTW-CS	1400	Logistic Regression	$84,76 \pm 1,29$
		Linear SVM	$84,34 \pm 0,91$
	1030	Linear	$73,43 \pm 1,04$
DTW-CS-ReliefF	1070	Logistic Regression	$79,00 \pm 1,14$
	1250	Linear SVM	$84,19 \pm 1,14$
	101	Linear	$70,46 \pm 2,07$
DTW-CS-LASSO	101	Logistic Regression	$76,76 \pm 1,51$
	99	Linear SVM	$78,88 \pm 2,07$
		Linear	$64,70 \pm 1,95$
DTW-CCS	25200	Logistic Regression	$81,70 \pm 1,14$
		Linear SVM	$81,29 \pm 1,21$
	19110	Linear	$62,70 \pm 1,83$
DTW-CCS-ReliefF	19180	Logistic Regression	$75,09 \pm 1,49$
	22190	Linear SVM	$81,40 \pm 1,15$
	65	**Linear**	$\mathbf{49,41 \pm 5,28}$
DTW-CCS-**LASSO**	66	Logistic Regression	$59,00 \pm 5,80$
	66	Linear SVM	$56,20 \pm 6,84$
		Linear	$93,83 \pm 0,65$
BoCF	31500	Logistic Regression	$94,33 \pm 0,57$
		Linear SVM	$94,49 \pm 0,42$
	11795	Linear	$94,33 \pm 1,03$
BoCF-**ReliefF**	**4200**	**Logistic Regression**	$\mathbf{94,41 \pm 0,73}$
	12225	Linear SVM	$93,83 \pm 1,31$
	702	Linear	$91,19 \pm 0,74$
BoCF-LASSO	449	Logistic Regression	$89,84 \pm 0,79$
	702	Linear SVM	$90,40 \pm 1,43$
		Linear	$63,47 \pm 3,33$
PR-BoCF	1400	Logistic Regression	$88,01 \pm 1,70$
		Linear SVM	$90,03 \pm 1,34$
	446	Linear	$59,53 \pm 5,05$
PR-BoCF-ReliefF	419	Logistic Regression	$78,37 \pm 3,15$
	419	Linear SVM	$77,84 \pm 3,25$
	345	Linear	$62,73 \pm 3,59$
PR-BoCF-LASSO	344	Logistic Regression	$83,84 \pm 3,31$
	20	Linear SVM	$83,34 \pm 6,52$

pyramid pooling holds enough information to classify binary shapes. Hence, our scheme can reveal meaningful patterns leading the classification system to discriminate groups using a fewer set of attributes as we expected.

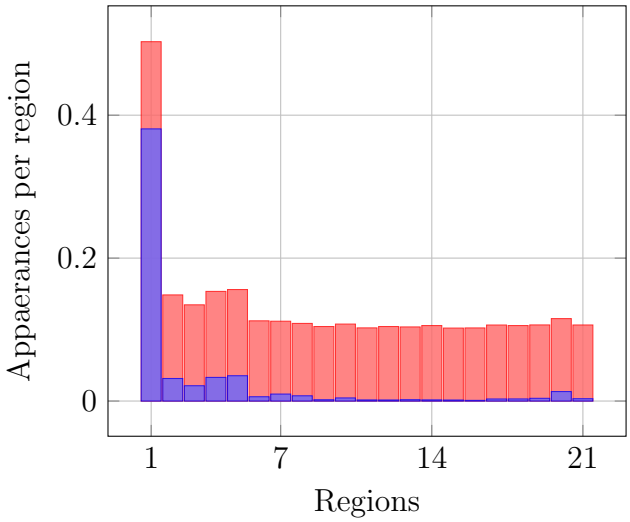

Figure 15: Histogram of selected feature by region for `MPEG7 DB`

The results presented in this section restate that our ensemble of FE, FR, and FS increases the IPS performance since it can discard non-informative features and to interpret the set of selected features as scales related to BI regions. Besides, our method shows that the inter-class variability favors its results because it allows discriminating with precision each DB class.

6.4.4 Feature Selection: PHGR Database

In this task of SC, we tried to recognize gestures from a hand gesture dataset. As we mentioned in Section 5.2, each sign is static, and each one corresponds to a specific hand gesture from the polish alphabet and numbers. Ought to that, the DB is homogeneous. Thus, a recognition system requires a set of proper descriptors to differentiate a sign from another. However, as Table 15 shows, our approach does not achieve outstanding performance because one of the enormous challenges of this dataset is its inter-class homogeneity.

In general, Table 15 note that our curvature-based approach is stronger than the `BoCF`-based one. Results from `BoCF` and PR-`BoCF` representations fail, since does not resolve well a class from another. Our parametric and non-parametric ways to represent CS sometimes achieve relevant outcomes. For instance, `DTW-CS` and `DTW-CCS` representations have an overall

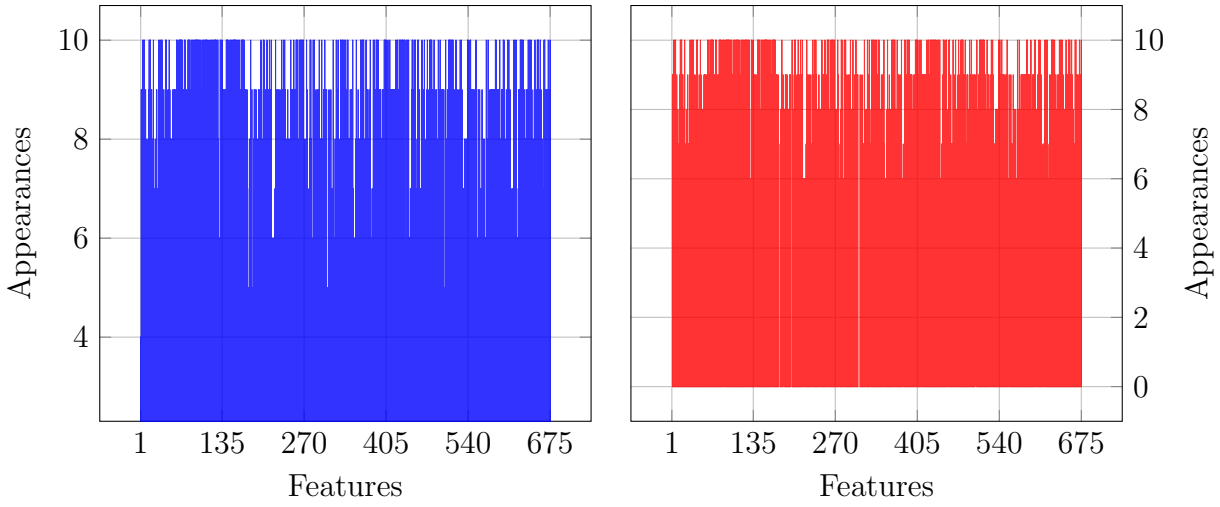

Figure 16: Histogram of features before and after filtering `PHGR DB`

performance that surpasses `BoCF` and PR-`BoCF`. Even when we applied FS methods (ReliefF and `LASSO`) over the original dataset, accuracy decay. It shows that our curvature-based scheme compacts relevant information to describe each category.

Once we discussed general remarks of results from Table 15, we will explain the best result. The discussion above notes that our best way to describe `PHGR DB` is the `DTW-CS` scheme applying the ReliefF FS method to get significant attributes from this dataset. Regarding the number of features selected, this method allows describing with fewer attributes than `BoCF`-based approaches to the classes involved in the problem. As a fact, the PR-`BoCF`-ReliefF ensemble cannot recognize each sign since it discards a good deal of information, leading to the identification of the static signs. Thus, the success procedure found a trade-off between a set of reduced features and acceptable accuracy outcomes.

Fig. 16 shows histograms of feature appearances before and after being filtered. These results indicate that as long as a DB has inter-class homogeneity, curvature-based features favor class discrimination. Yet, these attributes require a complementary resemblance like prototype representation, which codes similarities/dissimilarities between contour-based descriptors. Hence, to be `PHGR DB` well described, we need to preserve information as long as possible, even when we run FS. Both histograms hold recurrent "prototypes" instead of features since this representation comprises the attributes into the distance-based similarity computation. We preserve almost the whole DB, once we applied the ReliefF method.

In consequence, according to Table 15 and Fig. 16, similarity/dissimilarity-based representation approaches achieves satisfactory results over a homogeneous DB. Furthermore, our FE, FR, and FS methods reveal patterns that favor the IPS performance over the SSLR task.

Table 15: PHGR DB results for DTW-CS, DTW-CCS and BoCF representations and subsequent FS

Algorithm	Number of Features	Classifier	Accuracy%
		Linear	$58,43 \pm 2,49$
DTW-CS	675	Logistic Regression	$62,22 \pm 2,41$
		Linear SVM	$67,10 \pm 1,83$
	553	Linear	$57,99 \pm 2,22$
DTW-CS-ReliefF	568	Logistic Regression	$61,60 \pm 2,62$
	562	**Linear SVM**	$\mathbf{66,15 \pm 2,54}$
	45	Linear	$50,32 \pm 2,97$
DTW-CS-LASSO	47	Logistic Regression	$55,56 \pm 2,53$
	47	Linear SVM	$57,78 \pm 2,51$
		Linear	$60,38 \pm 1,71$
DTW-CCS	12150	Logistic Regression	$64,17 \pm 1,72$
		Linear SVM	$60,83 \pm 1,63$
	8100	Linear	$58,02 \pm 2,46$
DTW-CCS-ReliefF	6885	Logistic Regression	$62,60 \pm 2,51$
	9315	Linear SVM	$61,03 \pm 2,37$
	48	Linear	$40,18 \pm 4,26$
DTW-CCS-LASSO	46	Logistic Regression	$44,11 \pm 4,38$
	47	Linear SVM	$38,43 \pm 2,10$
		Linear	$37,40 \pm 2,46$
BoCF	31500	Logistic Regression	$38,40 \pm 2,74$
		Linear SVM	$37,16 \pm 2,60$
	4050	Linear	$38,03 \pm 1,99$
BoCF-ReliefF	9975	Logistic Regression	$38,73 \pm 2,69$
	9000	Linear SVM	$36,92 \pm 2,81$
	67	Linear	$30,06 \pm 1,46$
BoCF-LASSO	66	Logistic Regression	$32,90 \pm 1,97$
	67	Linear SVM	$30,21 \pm 2,45$
		Linear	$10,68 \pm 2,20$
PR-BoCF	675	Logistic Regression	$19,59 \pm 2,80$
		Linear SVM	$26,06 \pm 2,68$
	191	**Linear**	$\mathbf{9,70 \pm 1,78}$
PR-BoCF-ReliefF	236	Logistic Regression	$19,05 \pm 3,16$
	242	Linear SVM	$21,42 \pm 1,80$
	162	Linear	$10,24 \pm 2,85$
PR-BoCF-LASSO	162	Logistic Regression	$17,84 \pm 3,15$
	156	Linear SVM	$23,11 \pm 2,56$

6.4.5 Dimensionality Reduction and Visualization

Since we wanted to get more insights about the results showed in Section 6.4.1, Section 6.4.2, Section 6.4.3, and Section 6.4.4, we applied DR over the set of selected features get from FS stage. However, we divide it into two categories. First, DR to gather relevant attributes from a set of selected features, and second, DR for visualization. Hence, for the first DR approach, we use `CKA` [39] to get a lower space of features from the FS stage selected sets. Therefore, `CKA` allows us to reduce our attribute space, even more, revealing deeper information about the classes involved in the SC task.

Table 16: DR results using `CKA`

`KIMIA99SHAPE` DB			
Representation	**Number of Features**	**Classifier**	**Accuracy**%
`BoCF`-ReliefF	31	Linear	$97,20 \pm 2,23$
		Logistic Regression	$99,20 \pm 2,40$
		Linear SVM	$100,00 \pm 0,00$
`Animals` DB			
Representation	**Number of Features**	**Classifier**	**Accuracy**%
`BoCF`-ReliefF	783	Linear	$84,58 \pm 0,93$
		Logistic Regression	$85,75 \pm 1,07$
		Linear SVM	$85,05 \pm 1,14$
`MPEG7` DB			
Representation	**Number of Features**	**Classifier**	**Accuracy**%
`BoCF`-ReliefF	335	Linear	$94,29 \pm 0,75$
		Logistic Regression	$95,80 \pm 0,76$
		Linear SVM	$95,31 \pm 0,60$
`PHGR` DB			
Representation	**Number of Features**	**Classifier**	**Accuracy**%
`DTW-CS`-ReliefF	14	Linear	$37,31 \pm 2,85$
		Logistic Regression	$46,12 \pm 2,11$
		Linear SVM	$44,26 \pm 2,72$

Table 16 notes that for three `BoCF`-based representations and the features got, we keep performance results regarding those got in the FS phase. The DBs which posses inter-class and intra-class heterogeneity (`Kimia 99 Shape` DB, `Animals` DB, and `MPEG7` DB), using

BoCF-like selected attributes, when we applied CKA resolves this datasets. If we compare Table 16 results with Tables 10, 12 and 14 outcomes, accuracy remains almost the same. That is to say, FS keeps a considerable amount of redundant information, which needs to be pruned. Hence, CKA discard that dimensions representing redundancy

As a DR method, CKA reduces the input space. If we compute the percentage of features preserved concerning the entire input space, we find surprising results. The BoCF representation described well the three heterogeneous DBs. If we contrast the number of features concerning the whole input space we got: the number of descriptors got using CKA over Kimia 99 Shape DB is 0.1%, representing at the same time 3.4% of reduced space features for the same dataset. In Animals DB the output space of attributes represents 2.5% of the 31500 descriptors get using BoCF, an concerning the reduced version of the dataset we got 6.8%. The MPEG7 DB gets a proportion of dimensions of 1.1% of the original space, representing 8% of reduced representation after we applied FS over the dataset. As we mentioned, those results are astonishing, so the BoCF-based description of images clusters repetitive attributes. Therefore, our approach should be considered as a method to prune redundant information in BoCF, which not describes the datasets of BI.

In contrast, we found that for homogeneous databases like PHGR DB, the similarity/dissimilarity-based approaches resolve almost this kind of dataset. Even when we applied E FS, we got an overall performance about shape classification. Notwithstanding, when we use CKA over this DB, the performance comedown. This outcome could be associate with the representation used. As we mentioned in Section 5.2.2, the idea to depict a dataset using prototype-like representations relies on the capacity to code attribute information through metric computations. It means DTW allows us to encode similarities/dissimilarities between samples. Whereas, when we applied DR using CKA, we prune comparisons between samples, inducing to reject images that would allow discriminating each class. Thus, the DR method to reduce dimensions fails when the DB is almost uniform.

The second DR approach comprises a set of different LDR and NLDR methods to visualize each database, using as input the reduced spaces got trough FS stages. Figs. 17 to 20 displays each image from each dataset as a point. It depicts classes using distinct colors. Embeddings got for Kimia 99 Shape DB shows that tSNE, SNE, JSE, and NeRV, allows to visualize almost each class, since each point are proximate to its corresponding category. However, CKA and PCA does not resolve well each cluster, ought to some groups stack together. But, the set of visualization results notes that FS comprises enough information in HD to allow classifiers to discriminate almost every class.

Fig. 18 shows a complex DB since as we see the overall embeddings got displays stack data points. That is clear in JSE, NeRV, CKA and PCA spaces. Yet, tSNE and SNE depicts a substantial amount of groups. However, in Section 6.4.2 we mentioned that this DB has intra-class heterogeneity leading to break the local similarity. Hence, some inter-class

dissimilarities are diffuse since mixes different categories as it is clear in each embedding in Fig. 18. Then, group stacking is due to the diversity of the DB.

Fig. 19 notes a vast difference between `MPEG7` DB and `Animals` DB. Despite having more classes than the previous datasets, the embeddings got through `tSNE`, and `SNE` methods reveal that the reduced feature space describes almost the DB. Stacking is not a problem to deal with, ought to the intra-class and inter-class heterogeneity. These properties lead to a clear portrait of the categories involved, showing that our FS method depicts disparity among groups. But, `JSE`, `NeRV`, `CKA` and `PCA` methods fails, since stacks together samples. Thus, similarities/dissimilarities are not clear under these approaches.

Fig. 20 shows the LD outcomes once we applied DR methods. Here is apparent that neither embedding depicts this dataset. None of the six proposed methods for visualization could deal with inter-class homogeneity property (See Section 6.4.4). Therefore, the Fig. 20 shows that our FS approach fails to try to keep as much information as possible may get. Indeed, Table 16 notes that our proposed method does not solve the shape classification task for hand gesture recognition.

6.5 Summary

We implemented new stages into our IPS pipeline. We used novel FE and FR methods to get features from BI of different DBs. To represent those datasets we get `CCS` and `CS` sequences that we later represent trough similarity/dissimilarity-based methods. This procedure allows getting prototype-based and `BoCF`-based representations that increase the IPS scope. This approach introduces redundancy as we expose in Sections 6.4.1 to 6.4.3, which leads to overfitting. Hence, we applied E and NE FS techniques to prune non-informative features reducing the IPS complexity. The FS stage allows us to reveal scaled-based information related to the `BoCF` scheme, so we get interpretability. Yet, FS stages do not discard each redundant attribute. Then, we applied `CKA` as an NLDR method to get an LD feature space, increasing the IPS ability to keep useful features pruning those which do not have high variance. To increase the IPS interpretability, we implemented LDR and NLDR methods as visualization tools to gather the visual insight into our techniques. The results show that our methodology achieves 99.80 ± 0.60, 85.17 ± 0.95, and 94.41 ± 0.73 in `Kimia 99-Shape` DB, `Animals` DB, and `MPEG7` DB. However, when we applied our methods in SSLR DB, we get 66.15 ± 2.54. The outcomes, as mentioned earlier, show that our methodology performs well in heterogeneous DBs but not similar ones. But, our proposal's overall performance shows that FS and DR increase the IPS scope in BI SC tasks.

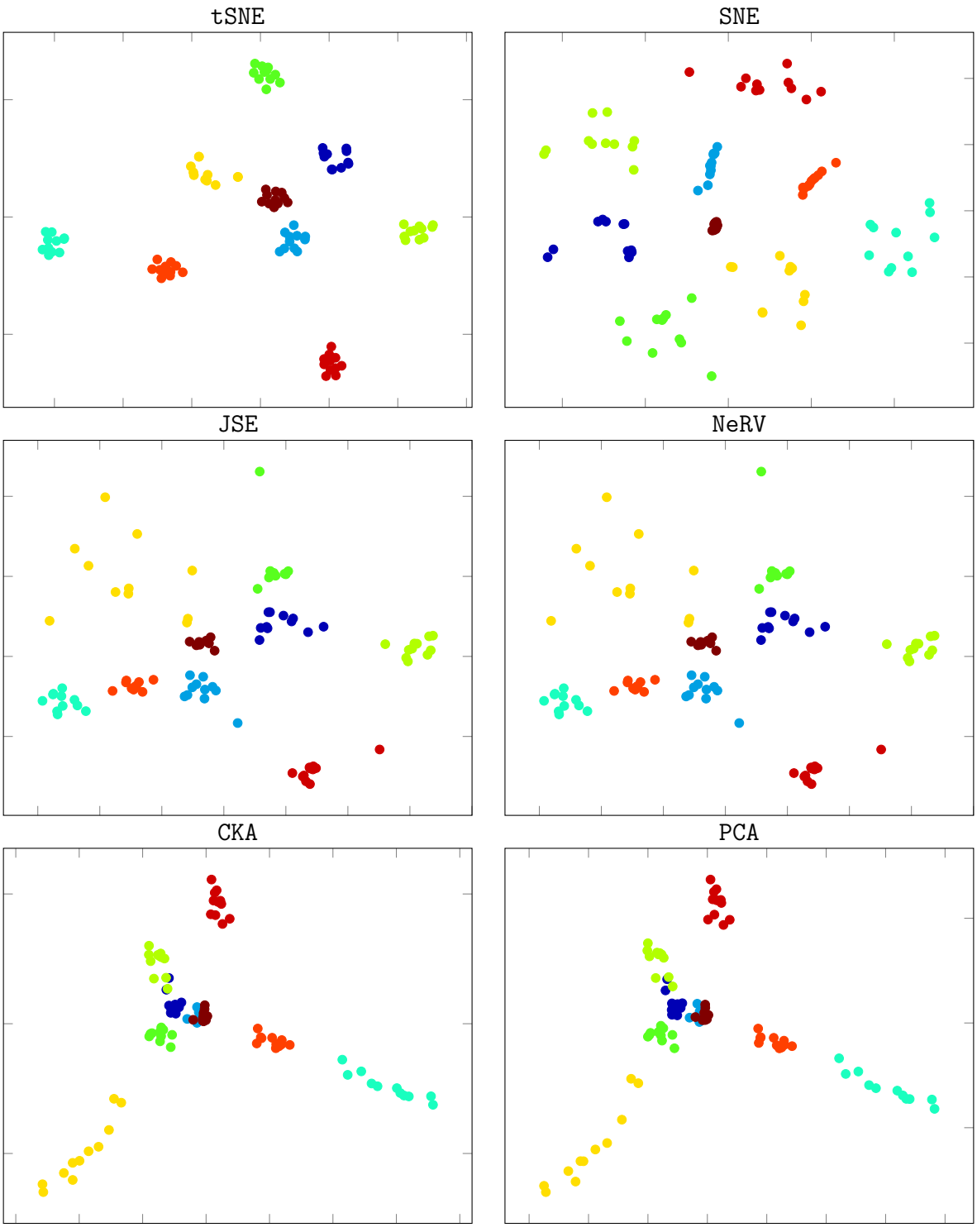

Figure 17: DR techniques for visualizing K99S DB

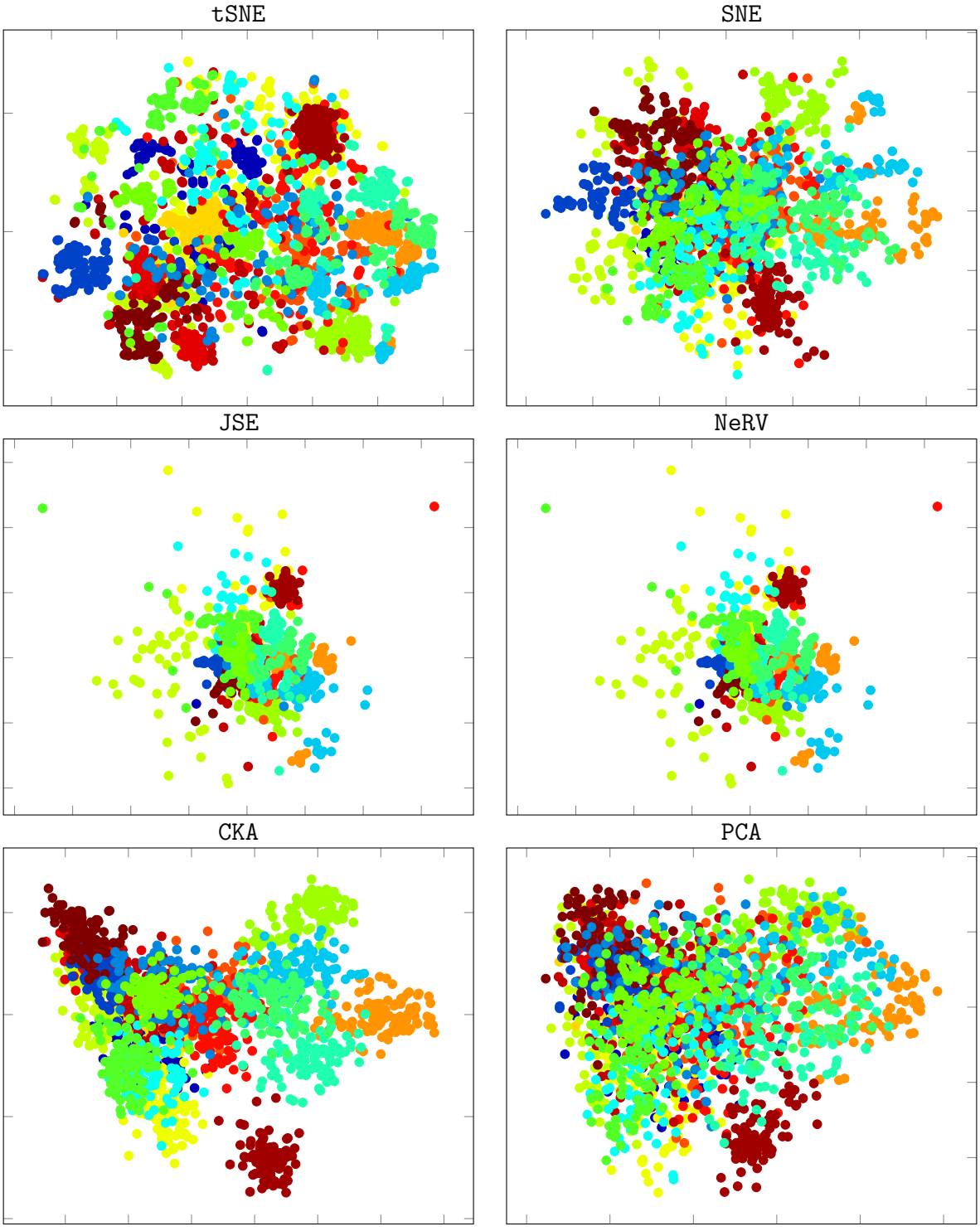

Figure 18: DR techniques for visualizing `Animals DB`

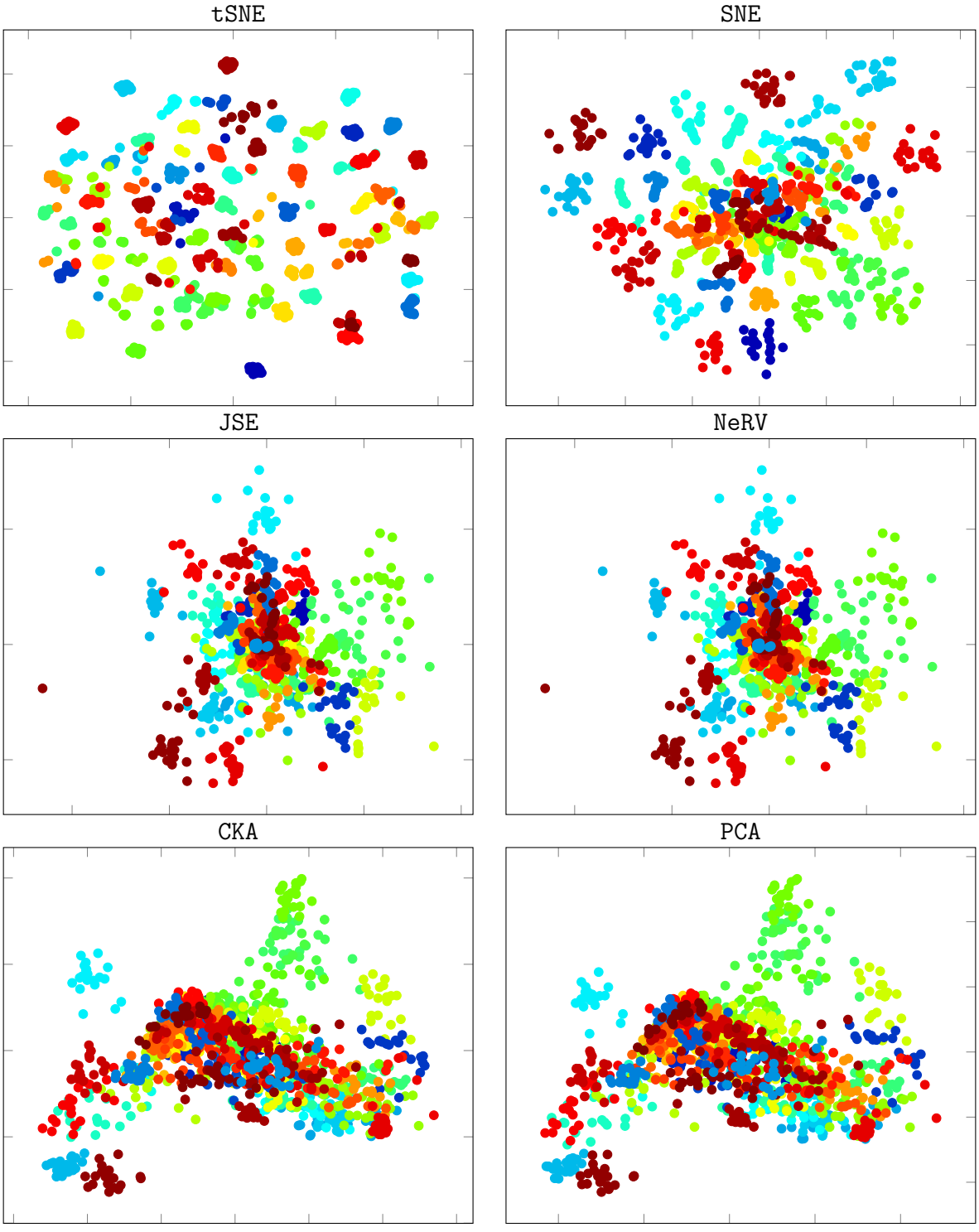

Figure 19: DR techniques for visualizing MPEG7 DB

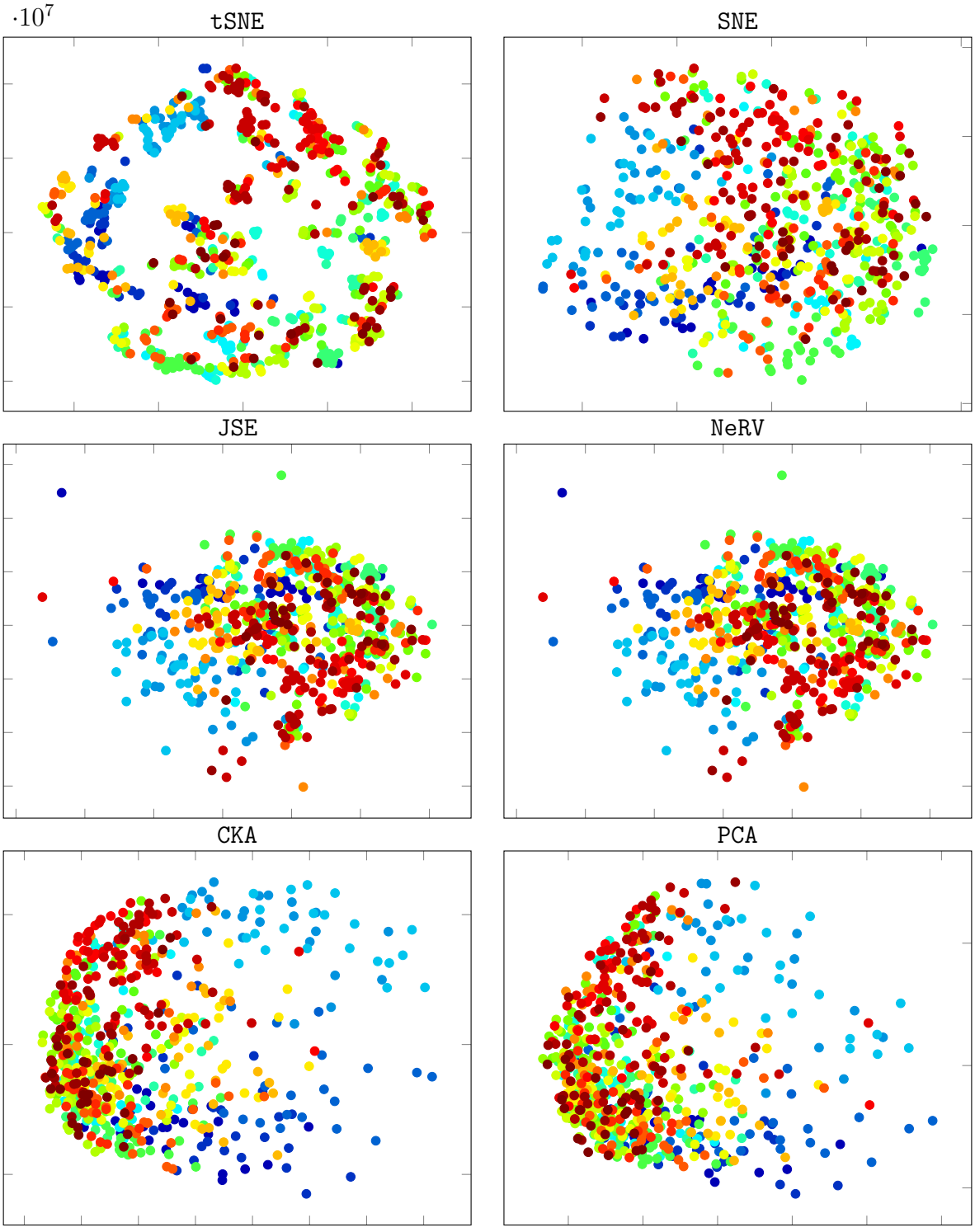

Figure 20: DR techniques for visualizing `PHGR DB`

7 Conclusions

To conclude, we develop an IPS based on similarity/dissimilarity representations to reveal relevant patterns in classification tasks. Our results expose that contour-based features are suitable to get shape information. Our IPS proves that when we filtered and reduced this information, we get a more in-depth insight into the DBs used. Even, we conclude that our findings unveil that scale-based approaches keep redundant features, which are low scales related. Thus, this system learns and preserves high-range data instead of the low one as the human perceives contours. Sometimes the feature space domain leads to overfitting. To avoid that, we include DR methods to reveal high-variance information, discarding those features that increase the IPS complexity. Our proposal shows performance results comparable to other State-Of-The-Art techniques. However, we noted that this occurs when DBs are heterogeneous. Instead, when the images present similar tendencies, our system learns from simple curvature approaches. We consider that using FE and FR phases based on contour-based features leads to get good SC results. Besides, FS and DR methods improve the IPS since prunes non-informative attributes and increase the interpretability of the results.

Regarding specific objective 1, we presented a novel approach to classify shapes based on sequential features extracted from contours and a `KAF`-based `MMD` enhancement. Thus, our approach captures non-linear curvature dependencies to code relevant shape patterns from BI. We proposed to support objects classification tasks over images, and our methodology achieves it, since we used this contour-based information to recognize shapes from `Kimia 99 Shape` DB and the sampled `MPEG7` DB. Our results are comparable concerning the other State-Of-The-Art techniques. However, `KAF`-based methods introduce a new mapping that is complex to get since it is based on kernel methods. These approaches are costly and slow to train. Then, this FR technique induces complexity to the IPS. Yet, our method finds a novel FE and FR stages of getting silhouette-based features. As we showed, these results can unveil a new manner of recognizing objects into an IPS system, showing that curvature-based features and similarity/dissimilarity-based representations can favor BI SC.

Concerning specific objective 2, we develop a novel DTW-based representation approach to support SSLR. As we did to achieve the specific objective 1, we estimated `CCS` coefficients from BI of hand gestures. We did that computing a multi-step length feature estimation to reveal both local and global variations towards shape edges. Then, we computed a `DTW`-based distance to compare `CCS` of different lengths. We used an E FS approach to code consistent patterns regarding the `DTW` features from different step lengths. Our proposal allows us to code morphologic similarities to varying scales between binary images using data sequences, encouraging discrimination among static signs. Attained results on a public database prove that we can achieve an 85% classification performance on average. Besides, to the best of our knowledge, this work is the first attempt to apply dissimilarity-based representations to codify sequential data from binary images devoted to SSLR. We used a simple set of

features and powerful representation techniques to learn data-driven classification models. The previous results show that one of the main problems of these kinds of methods is the inability to discriminate classes that share morphological structure. Hence, homogeneous DBs introduce a new challenge that our proposed method can not deal with yet. Therefore, the IPS needs to improve its SSLR capacity.

To achieve the specific objective 3, we assumed that the first two objectives led us to get a strong ensemble of FE and FR of contour-based attributes. Thus, we proved that silhouette-oriented approaches favor SC from BI. Their simplicity concerning other State-Of-The-Art techniques allows mapping these features to other spaces like similarity/dissimilarity-based. However, we develop a set of experiments showing the necessity to prune redundant information. In fact, we applied E and NE methods to test the methodology's ability to select relevant attributes from curvature-like descriptors (`CCS`, `CS`, and `BoCF`). The selection scheme filtered the negligible descriptors that might affect the DR methods applied that compute distances among samples. Another advantage of the proposed FS techniques is their classification complexity reduction, as we note in FS and `CKA`-oriented C stages. Overall, our methodology enhances the State-Of-The-Art results because we develop an FS strategy towards improving DR visualization and C of BI DBs. Therefore, this ensemble of FE, FR, FS, and DR techniques could increase the IPS performance because these methods improve the interpretability as long as the non-informative feature pruning.

8 Future Work

To increase the performance of the IPS, we proposed the possible future work according to each of the specific objectives proposed:

- **Specific Objective 1:** regarding the characterization methodology based on shape coding, we used `KAF`-based methods to learn models from shape sequences. However, its disadvantage relies on the σ_x optimization. Notwithstanding, authors in [52] proposed adaptive learning of the kernel bandwidth. Besides, another problem is to select the correct e_t value. Some authors, like [53], proposed to learn the step-size parameter as an online adaptive task. These methods would increase the FE stage scope.
- **Specific Objective 2:** concerning the similarity/dissimilarity-based strategy to represent contour-based features, some author has proposed to turn the shape sequences into matrices. Hence, one would have a matrix representation from each BI and classify it trough a similarity measure of matrices. Authors in [54] stated a method that implements a matrix-based euclidean distance and a `DTW` distance to measure the similarity between a set of matrices. Other authors in [55] proposed a novel metric to measure the misalignments between time series. This measure could be applied to our `CCS` and `CS` to get a new FR scheme that considers this problem since our sequences have distinct extents.
- **Specific Objective 3:** we proposed to develop a DR methodology to measure coding patterns from BI to favors C and visualization. However, some problems arise when we get HD spaces, and we want to prune redundant information. We try to solve this using NE FS methods. The `LASSO` method finds linear correlations between features and their class labels. However, some relations are not linear. Hence, authors in [56] proposed a kernelized `LASSO` which explores non-linear relations between features and labels. Besides, our scheme deals with a DR stage. We found that some methods do not include label information. Hence, we would explore the work from [57] that turns the `t-SNE` into a supervised DR method. This strategy is useful since the labels provide information about how the input data is spatially distributed.

9 Academic Results

We present the papers we published and those which have been submitted.

Table 17: Academic results

Objective	Category	Paper
• To develop a characterization methodology based on shape coding to support objects C tasks over BI	Q2	*Shape classification using Hilbert space embeddings and kernel adaptive filtering* [58]
• To develop a representation strategy based on similarity/dissimilarity measures, which quantifies similarities between regions of interest to code relevant patterns in BIs.	Q2	*A DTW-Based Representation to Support Static Sign Language Recognition from Binary Images* [59]
• To develop a DR methodology based on similarity/dissimilarity measures that allow coding relevant patterns in BI C and data visualization tasks in IPS.	Q1	*Paper send to Pattern Recognition Letters from Elsevier*

10 Acknowledgments

This work was fully funded by COLCIENCIAS projects: "Metodología para el reconocimiento y la traducción de señas aisladas en la Lengua de Señas Colombiana utilizando técnicas de visión por computador" and "Metodología para el reconocimiento y la traducción de señas aisladas en la Lengua de Señas Colombiana utilizando técnicas de visión por computador: Plan de formación - Metodología para el reconocimiento de señas aisladas en Lengua de Señas Colombiana utilizando características secuenciales, aprendizaje de máquina y técnicas de visión por computador". Also, this research was partially funded by the Vicerrectoría de Investigaciones, Innovación y Extensión project E6-18-09 from Universidad Tecnológica de Pereira.

Bibliography

[1] D. Lu and Q. Weng, "A survey of image classification methods and techniques for improving classification performance," *International journal of Remote sensing*, vol. 28, no. 5, pp. 823–870, 2007.

[2] Z. Guo, L. Zhang, and D. Zhang, "Rotation invariant texture classification using lbp variance (lbpv) with global matching," *Pattern recognition*, vol. 43, no. 3, pp. 706–719, 2010.

[3] N. R. Pal and S. K. Pal, "A review on image segmentation techniques," *Pattern recognition*, vol. 26, no. 9, pp. 1277–1294, 1993.

[4] N. Mohajeri, D. Assouline, B. Guiboud, A. Bill, A. Gudmundsson, and J.-L. Scartezzini, "A city-scale roof shape classification using machine learning for solar energy applications," *Renewable energy*, vol. 121, pp. 81–93, 2018.

[5] W. Yuan, K. Chin, M. Hua, G. Dong, and C. Wang, "Shape classification of wear particles by image boundary analysis using machine learning algorithms," *Mechanical Systems and Signal Processing*, vol. 72, pp. 346–358, 2016.

[6] F. Li, C. Shang, Y. Li, and Q. Shen, "Feature ranking-guided fuzzy rule interpolation for mammographic mass shape classification," in *2018 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*. IEEE, 2018, pp. 1–7.

[7] C. Keskin, F. Kıraç, Y. E. Kara, and L. Akarun, "Hand pose estimation and hand shape classification using multi-layered randomized decision forests," in *European Conference on Computer Vision*. Springer, 2012, pp. 852–863.

[8] Y. Rathi, N. Vaswani, A. Tannenbaum, and A. Yezzi, "Tracking deforming objects using particle filtering for geometric active contours," *IEEE transactions on pattern analysis and machine intelligence*, vol. 29, no. 8, pp. 1470–1475, 2007.

[9] C. Luo and L. Ma, "Manifold regularized distribution adaptation for classification of remote sensing images," *IEEE Access*, 2018.

[10] P. K. Pisharady and M. Saerbeck, "Recent methods and databases in vision-based hand gesture recognition: A review," *Computer Vision and Image Understanding*, vol. 141, pp. 152–165, 2015.

[11] H. Ling and D. W. Jacobs, "Shape classification using the inner-distance," *IEEE transactions on pattern analysis and machine intelligence*, vol. 29, no. 2, pp. 286–299, 2007.

[12] X. Bai, W. Liu, and Z. Tu, "Integrating contour and skeleton for shape classification," in *2009 IEEE 12th international conference on computer vision workshops, ICCV workshops*. IEEE, 2009, pp. 360–367.

[13] M. Bicego and V. Murino, "Investigating hidden markov models' capabilities in 2d shape classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 2, pp. 281–286, 2004.

[14] M. Bicego, V. Murino, and M. A. Figueiredo, "Similarity-based classification of sequences using hidden markov models," *Pattern Recognition*, vol. 37, no. 12, pp. 2281–2291, 2004.

[15] M. R. Daliri and V. Torre, "Shape recognition based on kernel-edit distance," *Computer Vision and Image Understanding*, vol. 114, no. 10, pp. 1097–1103, 2010.

[16] J. Wang, X. Bai, X. You, W. Liu, and L. J. Latecki, "Shape matching and classification using height functions," *Pattern Recognition Letters*, vol. 33, no. 2, pp. 134–143, 2012.

[17] X. Bai, C. Rao, and X. Wang, "Shape vocabulary: A robust and efficient shape representation for shape matching," *IEEE Transactions on Image Processing*, vol. 23, no. 9, pp. 3935–3949, 2014.

[18] J. J. d. M. S. Junior and A. R. Backes, "Shape classification using line segment statistics," *Information Sciences*, vol. 305, pp. 349–356, 2015.

[19] M. Bicego and P. Lovato, "A bioinformatics approach to 2d shape classification," *Computer Vision and Image Understanding*, vol. 145, pp. 59–69, 2016.

[20] X. Wang, B. Feng, X. Bai, W. Liu, and L. J. Latecki, "Bag of contour fragments for robust shape classification," *Pattern Recognition*, vol. 47, no. 6, pp. 2116–2125, 2014.

[21] W. Shen, X. Wang, C. Yao, and X. Bai, "Shape recognition by combining contour and skeleton into a mid-level representation," in *Chinese Conference on Pattern Recognition*. Springer, 2014, pp. 391–400.

[22] W. Shen, Y. Jiang, W. Gao, D. Zeng, and X. Wang, "Shape recognition by bag of skeleton-associated contour parts," *Pattern Recognition Letters*, vol. 83, pp. 321–329, 2016.

[23] W. Shen, W. Gao, Y. Jiang, D. Zeng, and Z. Zhang, "Shape recognition by bag of contour fragments with a learned pooling function," in *Image Processing (ICIP), 2017 IEEE International Conference on*. IEEE, 2017, pp. 1037–1041.

[24] W. Shen, C. Du, Y. Jiang, D. Zeng, and Z. Zhang, "Bag of shape features with a learned pooling function for shape recognition," *Pattern Recognition Letters*, 2018.

[25] B. Ramesh, C. Xiang, and T. H. Lee, "Shape classification using invariant features and contextual information in the bag-of-words model," *Pattern Recognition*, vol. 48, no. 3, pp. 894–906, 2015.

[26] C. M. Bishop, *Pattern recognition and machine learning*. springer, 2006.

[27] M. Dash and H. Liu, "Feature selection for classification," *Intelligent data analysis*, vol. 1, no. 3, pp. 131–156, 1997.

[28] K. Bunte and J. A. Lee, "Unsupervised dimensionality reduction: the challenges of big data visualization," in *Proceedings of the European Symposium on Artificial Neural Networks*, 2015, pp. 487–494.

[29] J. J. d. M. S. Junior, A. R. Backes, and O. M. Bruno, "Randomized neural network based descriptors for shape classification," *Neurocomputing*, vol. 312, pp. 201–209, 2018.

[30] J. Zeng, M. Liu, X. Fu, R. Gu, and L. Leng, "Curvature bag of words model for shape recognition," *IEEE Access*, vol. 7, pp. 57 163–57 171, 2019.

[31] M. Clément, C. Kurtz, and L. Wendling, "Enlacement and interlacement shape descriptors," in *International Conference on Pattern Recognition and Artificial Intelligence (ICPRAI)*, 2020.

[32] M. Müller, "Dynamic time warping," *Information retrieval for music and motion*, pp. 69–84, 2007.

[33] M. Robnik-Šikonja and I. Kononenko, "Theoretical and empirical analysis of relieff and rrelieff," *Machine learning*, vol. 53, no. 1-2, pp. 23–69, 2003.

[34] M. Miao, A. Wang, and F. Liu, "A spatial-frequency-temporal optimized feature sparse representation-based classification method for motor imagery eeg pattern recognition," *Medical & biological engineering & computing*, vol. 55, no. 9, pp. 1589–1603, 2017.

[35] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *J MACH LEARN RES*, vol. 9, no. Nov, pp. 2579–2605, 2008.

[36] G. E. Hinton and S. T. Roweis, "Stochastic neighbor embedding," in *Advances in neural information processing systems*, 2003, pp. 857–864.

[37] J. A. Lee, E. Renard, G. Bernard, P. Dupont, and M. Verleysen, "Type 1 and 2 mixtures of kullback–leibler divergences as cost functions in dimensionality reduction based on similarity preservation," *Neurocomputing*, vol. 112, pp. 92–108, 2013.

[38] J. Venna, J. Peltonen, K. Nybo, H. Aidos, and S. Kaski, "Information retrieval perspective to nonlinear dimensionality reduction for data visualization," *Journal of Machine Learning Research*, vol. 11, no. Feb, pp. 451–490, 2010.

[39] C. Cortes, M. Mohri, and A. Rostamizadeh, "Algorithms for learning kernels based on centered alignment," *Journal of Machine Learning Research*, vol. 13, no. Mar, pp. 795–828, 2012.

[40] H. S. Coxeter, "Introduction to geometry," 1989.

[41] B. K. Sriperumbudur, A. Gretton, K. Fukumizu, B. Schölkopf, and G. R. Lanckriet, "Hilbert space embeddings and metrics on probability measures," *Journal of Machine Learning Research*, vol. 11, no. Apr, pp. 1517–1561, 2010.

[42] C. D. Zuluaga, E. A. Valencia, M. A. Álvarez, and Á. A. Orozco, "A parzen-based distance between probability measures as an alternative of summary statistics in approximate bayesian computation," in *International Conference on Image Analysis and Processing*. Springer, 2015, pp. 50–61.

[43] B. Chen, S. Zhao, P. Zhu, and J. C. Principe, "Quantized kernel least mean square algorithm," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 23, no. 1, pp. 22–32, 2012.

[44] C.-M. Pun and C. Lin, "Geometric invariant shape classification using hidden markov model," in *Digital Image Computing: Techniques and Applications (DICTA), 2010 International Conference on*. IEEE, 2010, pp. 406–410.

[45] R. W. Schafer *et al.*, "What is a savitzky-golay filter," *IEEE Signal processing magazine*, vol. 28, no. 4, pp. 111–117, 2011.

[46] W. Liu, J. C. Principe, and S. Haykin, *Kernel adaptive filtering: a comprehensive introduction*. John Wiley & Sons, 2011, vol. 57.

[47] J. Nalepa and M. Kawulok, "Fast and accurate hand shape classification," in *IC BDAS 2014*. Springer, 2014, pp. 364–373.

[48] N. Cristianini, J. Shawe-Taylor, A. Elisseeff, and J. S. Kandola, "On kernel-target alignment," in *Advances in neural information processing systems*, 2002, pp. 367–373.

[49] H. Federer, "Curvature measures," *Transactions of the American Mathematical Society*, vol. 93, no. 3, pp. 418–491, 1959.

[50] V. Fonti and E. Belitser, "Feature selection using lasso," *VU Amsterdam Research Paper in Business Analytics*, pp. 1–25, 2017.

[51] R. Muthukrishnan and R. Rohini, "Lasso: A feature selection technique in predictive modeling for machine learning," in *2016 IEEE International Conference on Advances in Computer Applications (ICACA)*. IEEE, 2016, pp. 18–20.

[52] J. Zhao, H. Zhang, and J. A. Zhang, "Gaussian kernel adaptive filters with adaptive kernel bandwidth," *Signal Processing*, vol. 166, p. 107270, 2020.

[53] S. Garcia-Vega, X.-J. Zeng, and J. Keane, "Learning from data streams using kernel least-mean-square with multiple kernel-sizes and adaptive step-size," *Neurocomputing*, vol. 339, pp. 105–115, 2019.

[54] Y. Ye, J. Jiang, B. Ge, Y. Dou, and K. Yang, "Similarity measures for time series data classification using grid representation and matrix distance," *Knowledge and Information Systems*, vol. 60, no. 2, pp. 1105–1134, 2019.

[55] G. Jiang, W. Wang, and W. Zhang, "A novel distance measure for time series: Maximum shifting correlation distance," *Pattern Recognition Letters*, vol. 117, pp. 58–65, 2019.

[56] M. Yamada, W. Jitkrittum, L. Sigal, E. P. Xing, and M. Sugiyama, "High-dimensional feature selection by feature-wise kernelized lasso," *Neural computation*, vol. 26, no. 1, pp. 185–207, 2014.

[57] C. de Bodt, D. Mulders, D. López-Sánchez, M. Verleysen, and J. A. Lee, "Class-aware t-sne: cat-sne," in *European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, 2019.

[58] J. Blandon, C. K. Valencia, A. Alvarez, J. Echeverry, M. Alvarez, and A. Orozco, "Shape classification using hilbert space embeddings and kernel adaptive filtering," in *International Conference Image Analysis and Recognition*. Springer, 2018, pp. 245–251.

[59] J. Blandon, A. Alvarez, and A. Orozco, "A dtw-based representation to support static sign language recognition from binary images," in *Iberoamerican Congress on Pattern Recognition*. Springer, 2018, pp. 938–945.